

Trabajo Fin de Grado

Grado en Ingeniería de Tecnologías Industriales

Automatización de un Sistema de Ascensores

Autor: Daniel Paredes García

Tutor: Francisco Rodríguez Rubio

**Dep. Ingeniería de Sistemas y Automática
Escuela Técnica Superior de Ingeniería
Universidad de Sevilla**

Sevilla, 2017



Trabajo Fin de Grado
Grado en Ingeniería de Tecnologías Industriales

Automatización de un Sistema de Ascensores

Autor:

Daniel Paredes García

Tutor:

Francisco Rodríguez Rubio

Catedrático de Universidad

Dep. Ingeniería de Sistemas y Automática

Escuela Técnica Superior de Ingeniería

Universidad de Sevilla

Sevilla, 2017

Trabajo Fin de Grado: Automatización de un Sistema de Ascensores

Autor: Daniel Paredes García

Tutor: Francisco Rodríguez Rubio

El tribunal nombrado para juzgar el Proyecto arriba indicado, compuesto por los siguientes miembros:

Presidente:

Vocales:

Secretario:

Acuerdan otorgarle la calificación de:

Sevilla, 2017

El Secretario del Tribunal

A mi familia

Agradecimientos

Me gustaría aprovechar estas líneas para recordar a todas aquellas personas que me han apoyado y han estado conmigo durante estos cuatro años.

En primer lugar, agradecer a mis padres todo el apoyo, el cariño y la confianza que me han dado, no solo en esta etapa universitaria, sino durante toda mi vida. Siempre han creído en mí y, tanto en las buenas como en las malas, los he tenido siempre conmigo.

Agradecer también a mi tutor Francisco Rodríguez la oportunidad que me ha brindado para llevar a cabo este TFG, y en especial al profesor Fernando Luis Castaño quien, de manera desinteresada, me ha ayudado a poder sacarlo adelante.

A Virginia, la persona que más me ha sufrido durante este tiempo, alegrándose incluso más que yo por mis logros y animándome y estando a mi lado cuando más la necesitaba. Gracias por darme tanto a cambio de tan poco.

Y por último, a mi primo Álvaro, mi “profesor particular y principal proveedor de apuntes”. Gracias a ti obtuve hasta un reconocimiento. Hemos vivido grandes momentos durante estos años en clase, en los trabajos o en las prácticas. Hemos tenido algún que otro roce pero se quedan en nada comparado con las miles de risas y buenos momentos. Hace 4 años empezamos juntos la carrera siendo primos y amigos. Hoy, 4 años después, la terminamos juntos, convirtiéndote en un hermano para mí. Gran parte de mi éxito te lo debo a ti.

Daniel Paredes García

Sevilla, 2017

Resumen

Se pretende realizar la simulación y posterior implementación en una maqueta de un sistema de ascensores. Tanto la simulación como la maqueta contarán con dos ascensores, pero el número de plantas variará, ya que los ascensores de la simulación se diseñarán con siete pisos, mientras que los de la maqueta real cuentan solamente con cuatro plantas.

Para realizar el control de los ascensores se programarán dos tipos de maniobras diferentes con el fin de recrear distintas situaciones en función del ámbito donde sea requerida la instalación de un ascensor. Así pues, una de ellas será la maniobra universal, la cual tan solo atiende una petición por ascensor y trayecto, quedando el ascensor en exclusividad para el usuario que ha realizado la petición. Por otro lado, se programará la maniobra colectiva selectiva en subida y bajada, que permite atender todo tipo de llamadas realizadas, siempre y cuando sean peticiones para realizar un movimiento en el mismo sentido de desplazamiento del ascensor. Con la primera maniobra se intenta recrear un típico sistema de ascensores de un bloque de pisos donde el tráfico de personas es escaso; mientras que la segunda es una maniobra muy utilizada, por ejemplo, en edificios de oficinas.

La ejecución de las maniobras se llevará a cabo mediante los softwares informáticos CoDeSys y Unity Pro, que serán los encargados de realizar la simulación y la implementación, respectivamente. Para que el programa que controla la maqueta funcione correctamente, la maqueta será adaptada para que pueda ser utilizada con uno de los autómatas disponibles en la Escuela, el Modicon M340, por lo que será necesario realizar una instalación y conexionado previos de la maqueta con el autómata.

La ubicación tanto de la maqueta como del autómata es el laboratorio del departamento de Ingeniería de Sistemas y Automática, siendo la maqueta componente de un antiguo PFC.

Abstract

It is intended to carry out the simulation and subsequent implementation in a scale model of an elevator system. Both the simulation and the model will have two lifts, but the number of floors will vary, since the lifts of the simulation will be designed with seven floors, while those of the real model will only have four floors.

To control the lifts, two types of different maneuvers will be programmed to recreate different situations depending on the area where the installation is required. Thus, one of them will be the universal maneuver, which only attends a request for lift and journey, leaving the lift in exclusivity for the user who has made the request. On the other hand, selective collective maneuver will be programmed on the way up and down, which allows to attend all types of calls made, as long as they are requests to make a move in the same direction of elevator movement. The first maneuver attempts to recreate a typical system of lifts in a block of flats where people traffic is scarce; while the second one is a used maneuver, for instance, in office buildings.

The execution of the maneuvers will be carried out using the computer software CoDeSys and Unity Pro, which will be in charge of performing the simulation and implementation, respectively. In order for the program that controls the model works correctly, the model will be adapted so that it can be used with one of the PLCs available in the School, the Modicon M340, which will require a previous installation and connection of the model with the automaton.

The location of both the model and the PLC is the laboratory of the Department of Systems Engineering and Automation, being the model component of an old PFC.

Índice

Agradecimientos	ix
Resumen	xi
Abstract	xiii
Índice	xv
Índice de Tablas	xvii
Índice de Figuras	xix
1 Introducción	1
1.1 Objetivo	1
1.2 Estructura del trabajo	1
2 Historia de los sistemas de elevación	3
2.1 Reseña histórica	3
2.1.1 Antigüedad	3
2.1.2 Edad Contemporánea	4
2.2 Actualidad	5
2.2.1 Clasificación por Sistema de Tracción	5
2.2.1.1 Ascensores electromecánicos	5
2.2.1.2 Ascensores autoportantes	6
2.2.1.3 Ascensores hidráulicos	7
2.2.2 Control de maniobra de los ascensores	7
2.2.2.1 Maniobra universal	7
2.2.2.2 Maniobra colectiva en bajada	8
2.2.2.3 Maniobra colectiva selectiva en ambos sentidos de la marcha	8
3 Ascensor electromecánico	9
3.1 Sala de máquinas	10
3.1.1 Grupo tractor	10
3.1.1.1 Motor	10
3.1.1.2 Freno	10
3.1.1.3 Reductor	11
3.1.1.4 Polea de tracción	11
3.1.1.5 Volante de inercia	12
3.1.2 Limitador de velocidad	12
3.1.3 Cuadro de maniobra	12
3.2 Hueco del ascensor	13
3.2.1 Cabina	13
3.2.1.1 Bastidor	13
3.2.1.2 Caja	13
3.2.2 Guías y fijación de la cabina	13
3.2.3 Contrapeso	14
3.2.4 Guías y fijaciones del contrapeso	14
3.2.5 Cables	14
3.2.6 Paracaídas	15

3.3	Foso	15
3.3.1	Amortiguadores	15
3.3.2	Interruptores de seguridad	16
4	Maqueta de ascensores y autómatas	17
4.1	Maqueta del ascensor	17
4.2	Autómata programable	20
4.2.1	Bastidor BMX XBP 0400	20
4.2.2	Módulo de alimentación BMX CPS 2000	21
4.2.3	Procesador BMX XBP 0400	21
4.2.4	Módulo BMX DDI 3202K	22
4.2.5	Módulo BMX DDM 16022	23
4.3	Otros elementos	25
4.4	Conexionado	27
5	CoDeSys	33
5.1	Introducción al Software	34
5.2	Simulación de dos ascensores de 7 plantas con maniobra universal	36
5.2.1	Lenguaje SFC	37
5.2.2	Lenguaje LD	39
5.2.3	Lenguaje ST	39
5.3	Simulación de dos ascensores de 7 plantas con maniobra colectiva selectiva	42
5.3.1	Lenguaje SFC	42
5.3.2	Lenguaje LD	44
5.3.3	Lenguaje ST	44
6	Unity Pro	49
6.1	Introducción al Software	49
6.2	Maqueta de dos ascensores de 4 plantas con maniobra universal	53
6.3	Maqueta de dos ascensores de 4 plantas con maniobra colectiva selectiva	54
7	Conclusiones y Líneas de mejora	57
7.1	Líneas de mejoras	57
7.2	Conclusiones	57
ANEXO I: SIMULACIÓN MANIOBRA UNIVERSAL		59
ANEXO II: MAQUETA MANIOBRA UNIVERSAL		66
ANEXO III: SIMULACIÓN MANIOBRA COLECTIVA SELECTIVA		72
ANEXO IV: MAQUETA MANIOBRA COLECTIVA SELECTIVA		83
Referencias		99
Glosario		101

ÍNDICE DE TABLAS

Tabla 4.1 Correlación cables-conectores del 1º DB 25	18
Tabla 4.2 Correlación cables-conectores del 2º DB 25	19
Tabla 4.3 Relación sensores-colores del BMX DDI 3202K	22
Tabla 4.4 Relación sensores-colores del BMX DDM 16022	24
Tabla 4.5 Conexionado de cables de las entradas del 1º DB con el autómata	27
Tabla 4.6 Conexionado de cables de las entradas del 2º DB con el autómata	28
Tabla 4.7 Conexión de los motores con sus terminales correspondientes	28
Tabla 4.8 Conexión de los leds con sus terminales correspondientes	29
Tabla 4.9 Conexionado de las salidas	29
Tabla 6.1 Características básicas de Unity Pro	49
Tabla 6.2 Variables diferentes o añadidas respecto a la simulación de la maniobra universal	54
Tabla 6.3 Variables diferentes o añadidas respecto a la simulación de la maniobra colectiva selectiva	55

ÍNDICE DE FIGURAS

Figura 2.1 Elevación de agua mediante el shadoof	3
Figura 2.2 Ascensor ‘Teagle’ de 1845	4
Figura 2.3 Demostración del primer ascensor de Otis en el Palacio de Cristal de Nueva York	4
Figura 2.4 Ascensore Electrmecánico	6
Fifura 2.5 Ascensor Autoportable	6
Figura 2.6 Ascensor Hidráulico	7
Figura 3.1 Esquema de un ascensor electromecánico con sala de máquinas	9
Figura 3.2 Grupo Tractor	10
Figura 3.3 Interior del reductor de un grupo tractor	11
Figura 3.4 Polea de tracción de un grupo tractor	11
Figura 3.5 Limitador de velocidad	12
Figura 3.6 Cuadro de maniobras	13
Figura 3.7 Cabina de ascensor	13
Figura 3.8 Contrapeso de un ascensor	14
Figura 3.9 Amortiguador del foso de un ascensor	16
Figura 4.1 Maqueta ascensor utilizada en el proyecto	19
Figura 4.2 Autómata programable Modicon M340	20
Figura 4.3 Esquema de la alimentación y puesta a tierra del BMX CPS 2000	21
Figura 4.4 Diagrama de conexión del módulo BMX DDI 3202K	23
Figura 4.5 Diagrama de conexión del módulo BMX DDM 16022	24
Figura 4.6 Esquema de la conexión del motor con los relés	25
Figura 4.7 Esquema de la conexión de los leds con los relés	26
Figura 4.8 Circuito de conexionado del autómata	30
Figura 4.9 Esquema general de conexionado maqueta-autómata	31
Figura 5.1 Ventana principal del programa CoDeSys	34
Figura 5.2 Menú selección de autómata	35
Figura 5.3 Menú de configuración de la POU principal	35
Figura 5.4 A la izquierda, ventana de creación de una nueva POU. A la derecha, ventana de creación de una visualización	36
Figura 5.5 Grafcet general del Ascensor 1 con maniobra universal	38
Figura 5.6 Imagen de la visualización de los ascensores con maniobra universal	40

Figura 5.7 Variables globales utilizadas en la simulación de los ascensores con maniobra universal	41
Figura 5.8 Graficet general del Ascensor 1 con maniobra colectiva selectiva	43
Figura 5.9 Variables globales utilizadas en la simulación de los ascensores la maniobra colectiva selectiva (1)	46
Figura 5.10 Variables globales utilizadas en la simulación de los ascensores la maniobra colectiva selectiva (2)	47
Figura 6.1 Ventana principal del programa Unity Pro	50
Figura 6.2 Ventana del BUS PLC utilizado sin la elección de los módulos de E/S	50
Figura 6.3 Menú de selección de módulos de E/S	51
Figura 6.4 Ventana del BUS PLC con la elección de los módulos de E/S realizadas	51
Figura 6.5 Menú de configuración de una POU	52
Figura 6.6 Tabla de animación utilizada en los ascensores con maniobra colectiva selectiva	56

1 INTRODUCCIÓN

El hecho de que el precio del suelo se encarezca cada vez más debido a la masiva construcción de edificios en los últimos tiempos, obliga tanto a los ingenieros como a los arquitectos a buscar alternativas para paliar las diferentes dificultades que se les presenta. Esto ha hecho que las miras se centren en el cielo, es decir, en el aprovechamiento del espacio aéreo para poder seguir construyendo.

Cada vez son más las grandes edificaciones de cientos de plantas en las ciudades. Un ejemplo de ello es Sevilla donde, durante siglos, La Giralda que, con sus 101 metros de altura la hacía el monumento más alto de la ciudad. Su altura no podía ser superada por la construcción de ningún otro edificio, entre otras cosas, para poder visualizarse desde cualquier punto de la ciudad. Hoy en día, La Giralda ha perdido este honor en favor de la Torre Pelli, renombrada Torre Sevilla, cuya altura de 180 metros la convierte en el primer rascacielos de Sevilla. Con esto, se observa con claridad como los tiempos cambian y existe la necesidad de construir edificios cada vez más altos. Pero, ¿cómo llegar hasta estos espacios edificadas? La respuesta es muy sencilla: utilizando ascensores.

Según la RAE, un ascensor es un aparato que sirve para trasladar personas de unos pisos a otros [1]. Así pues, para poder trasladarnos entre las distintas plantas de un edificio sin ningún tipo de esfuerzo, tan sólo será necesario la instalación de uno, o varios, ascensores.

1.1 Objetivo

En lo que se refiere a este proyecto, el objetivo será la simulación e implementación en una maqueta que nos permita observar los diferentes sistemas de control de maniobra existentes en un ascensor en función de la localización y el flujo de personas del mismo. Para poder implementar este sistema de control a la maqueta deberá adaptarse dicha maqueta para poder ser utilizada con los distintos medios de los que se disponen en la Escuela, pasando a ser controladas por un autómata programable. Una vez adaptada dicha maqueta, podrá utilizarse en un futuro para diferentes fines educativos.

Para realizar la simulación se ha utilizado el software informático CoDeSys, el cual nos permite realizar simulaciones bastante reales para diferentes sistemas de control. Por su parte, para ser implementado en la maqueta, se ha necesitado el uso del software Unity Pro, ya que es el programa compatible con el autómata programable del que se dispone.

1.2 Estructura del trabajo

Este Trabajo Fin de Grado se encuentra estructurado de la siguiente manera:

El Capítulo 2, titulado ‘Historia de los sistemas de elevación’, contendrá un breve repaso a la evolución de los distintos medios de elevación realizados por el ser humano a lo largo de la historia llegando hasta la actualidad. Además, incluirá información sobre los distintos tipos de ascensores que se utilizan actualmente y las diferentes formas de control que los rigen.

En el capítulo 3, ‘Ascensor electromecánico’, se realizará una explicación más detallada de los elementos que componen este tipo de ascensor y su funcionamiento, ya que la maqueta utilizada para este Trabajo recrea un ascensor con estas características.

El capítulo 4, el cual se titula ‘Maqueta de ascensores y autómatas’, recoge toda la información sobre la maqueta utilizada, así como del autómata programable y todos los elementos utilizados. Además de esto, se realizará una explicación del sistema de conexionado que permite el uso de la maqueta junto con el autómata.

En el capítulo 5, denominado ‘CoDeSys’, se incluye una introducción a este programa informático. Además de esto, en él se recoge la explicación del modo en el cual ha sido programada la simulación de los diferentes sistemas de control que se han creado en este proyecto.

El capítulo 6, ‘Unity Pro’ incluye prácticamente la misma información que el capítulo 5 pero orientado a este nuevo software, es decir, contiene una breve introducción para el uso del programa y una explicación de cada uno de los sistemas de control utilizados en este proyecto en función de lo que ha sido programado.

El capítulo 7, titulado con el nombre ‘Conclusiones y líneas de mejora’ se recogen tanto las conclusiones a las que se ha llegado con la realización de proyecto como las diferentes líneas que pueden ser mejoradas en la maqueta o en el sistema de control.

Además de estos capítulos, se añaden 4 anexos que incluyen toda la codificación utilizada que permitirán la simulación y el control de la maqueta con los dos sistemas de control utilizados.

2 HISTORIA DE LOS SISTEMAS DE ELEVACIÓN

Antes de entrar en materia, se comenzará dedicando este apartado a explicar la evolución histórica de los diferentes sistemas de elevación, desde aquellos más primitivos que se usaban en la Antigüedad hasta los más modernos sistemas de ascensores que encontramos en la actualidad.

2.1 Reseña histórica

2.1.1 Antigüedad

Desde los inicios de la Humanidad, los humanos han sentido la necesidad de elevar diferentes cargas, ya fueran para crear construcciones o por simple necesidad, como pudiera ser extraer agua de los pozos. Así pues, el primer ejemplo apreciable donde se realizaban alzamientos de elevadas cargas data del Antiguo Egipto. En él, destacan las primeras pirámides egipcias, cuyas construcciones tenían más de 100 metros de altura y se utilizaban piedras de varias toneladas. Una de las teorías más extendidas, aunque a su vez poco probable, era el uso de rampas, pero está demostrado que hubiesen sido incluso más difíciles de construir que la propia pirámide.

Basándonos en hechos de los que se tienen constancia de la metodología utilizada para realizar elevaciones, debemos remontarnos allá por el año 1550 a.C., en los territorios de Egipto y Mesopotamia. Allí surgió el shadoof, un mecanismo de palanca que se utilizaba para extraer el agua procedente de los ríos mediante elevaciones de una cesta sujeta en un extremo de una extensa rama.

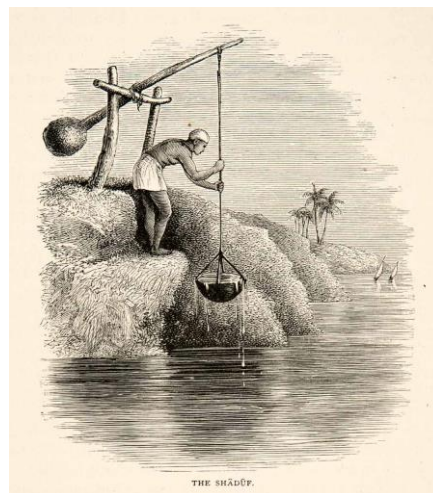


Figura 2.1 Elevación de agua mediante el shadoof

El primer elevador que se conoce se le otorga a Arquímedes en el año 236 a.C. el cual con el uso de una cuerda y una polea permitía, utilizando la fuerza ejercida por el hombre tirando desde un extremo de la cuerda, que la carga situada en el otro extremo se elevase.

En la Antigua Roma se utilizó este método para elevar a personas. El emperador Nerón utilizó este sistema en su palacio. Éste consistía en una cabina de madera de sándalo suspendida por un cable de cáñamo y guiada por cuatro carriles de madera. Gracias a la fuerza de los esclavos se elevaba hasta los pisos superiores. Ya que carecía de frenos se colocaba un cojín de un metro de espesor unido al suelo de la cabina.

Poco después, en la época del emperador Tito comenzaron a usarse estos montacargas en el Coliseo romano para elevar a los gladiadores y a los animales [2].

Durante los siglos posteriores la evolución de este sistema no evolucionó de forma considerable y solo se produjo una mejora de la técnica, sustituyéndose la fuerza del ser humano por la tracción animal. Posteriormente, se produciría la introducción los contrapesos.

2.1.2 Edad Contemporánea

Con la aparición de la máquina de vapor, creada por James Watt, comenzó a considerarse la idea de implantarla en los sistemas de elevación, permitiendo así que la elevación pudiese ser realizada por una máquina. De estos ascensores se puede destacar el ascensor “Teagle” desarrollado en Inglaterra en 1845. Dicho ascensor era accionado hidráulicamente y poseía unos contrapesos que permitían equilibrar la carga. El problema residía en que, en caso de rotura de los cables, era imposible detener la plataforma ya que carecía de un sistema de frenado.

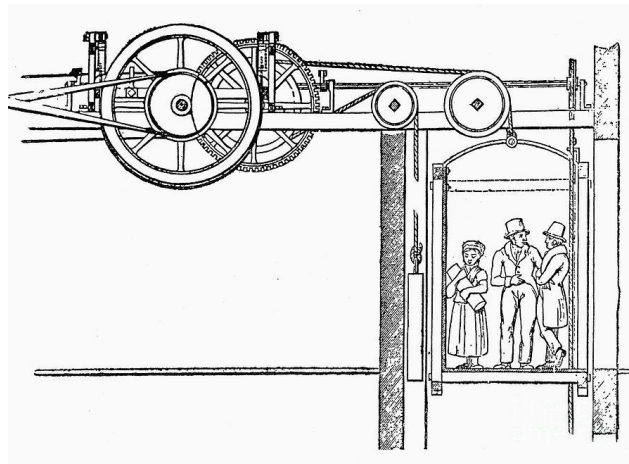


Figura 2.2 Ascensor ‘Teagle’ de 1845

Fue entonces cuando apareció la figura de **Elisha Graves Otis** quien, en 1852, creó un dispositivo de seguridad que aseguraba la integridad de las piezas en caso de rotura. A lo largo de la pared del hueco del elevador discurría una barra vertical dentada lo cual permitía, si se podrujera alguna rotura de los cables del ascensor, que la plataforma se clavase en dichos dientes y detendría la caída del dicho elevador [3].

En 1854 realizó una demostración en el Palacio de Cristal de Nueva York, situando el ascensor a una altura bastante considerable y haciendo cortar los cables que lo sustentaban para demostrar su sistema de seguridad. Esta demostración fue un éxito y solo tres años más tarde, en 1857 se instaló el que es considerado como el primer ascensor para personas, en *los Grandes Almacenes E.V. Haughwouth & Co.* (Nueva York), un edificio de cinco plantas donde el ascensor tenía una velocidad de ascenso de 12 metros por minuto [4] [5].



Figura 2.3 Demostración del primer ascensor de Otis en el Palacio de Cristal de Nueva York

A partir de este hecho, comenzaron a realizarse avances en la técnica y el modo de accionar las cabinas de los ascensores. Así, unos años más tarde, en 1867, el francés Leon Edoux presentó en la Exposición Universal de París el primer ascensor hidráulico, el cual utilizaba la presión del agua para elevar la cabina. En 1880, en la Exposición de Mannheim, Siemens expuso un ascensor accionado mediante electricidad. Sin embargo, no fue hasta 1889 cuando se construye el primer ascensor accionado por un motor eléctrico de corriente continua y con reductor, en el Demarest Building de Nueva York, desarrollado por Norton Otis, hijo de Elisha G. Otis [6].

A medida que la electricidad se iba extendiendo por el mundo, los ascensores eléctricos iban tomando el relevo a los ascensores hidráulicos e iban perfeccionándose. En 1903 se construye el primer ascensor eléctrico sin reductor con polea de tracción, que permitía transportar a los pasajeros a mayores alturas, llegando a alcanzar los 100 metros. Con las continuas mejoras con el paso de los años y con el sistema Ward Leonard que con su grupo motor-generator-motor permitió alcanzar grandes velocidades, se ha ido avanzando hasta llegar a los motores de hoy en día. Con estos motores y con los avances en los sistemas de seguridad se ha conseguido alcanzar los actuales ascensores fiables, innovadores y de gran calidad [7].

2.2 Actualidad

En la actualidad, numerosas empresas del sector se han encargado de ir mejorando y perfeccionando la técnica referida a la fabricación de los ascensores. Entre estas empresas destacan, como las más afincadas en España, *Otis Elevator Company*, impulsada por los hijos del propio Elisha Graves Otis y que continúa hasta nuestros días, o *Schindler Holding S.A.*, pionera en escaleras mecánicas en el mundo y entre las más populares en el ámbito de los ascensores.

Aunque todas las compañías tienden a mejorar sus ascensores en temas actuales como el ahorro energético, la contaminación acústica, la seguridad o la velocidad, todos los ascensores comparten las mismas técnicas en lo se refiere al sistema de tracción y al control de maniobra de los ascensores.

2.2.1 Clasificación por Sistema de Tracción

En función del sistema de tracción que utilizan los ascensores, se pueden destacar tres tipos diferentes: ascensores electromecánicos, ascensores hidráulicos y ascensores autoportantes o sin sala de máquinas.

2.2.1.1 Ascensores electromecánicos

Están compuestos por una cabina y un contrapeso. La masa del contrapeso debe ser igual a la de la cabina vacía más un 50% de la carga máxima que el motor es capaz de transportar.

Por su parte, la tracción se realiza por un motor eléctrico que permite subir y bajar gracias a unas guías y carriles fijos. Se compone, además, de una máquina reductora y una polea de la que cuelga el cable de tracción. Para la instalación de estos componentes es necesario una sala de máquinas, la cual suele estar instalada en las azoteas o parte superior del edificio.

Como sistema de seguridad estos ascensores poseen un amortiguador en la parte inferior del hueco del ascensor y un limitador de velocidad, que detiene el ascensor en caso de que se exceda la velocidad máxima.

Este tipo de ascensores es el más utilizado en viviendas multifamiliares y en edificios superiores a seis pisos [8].

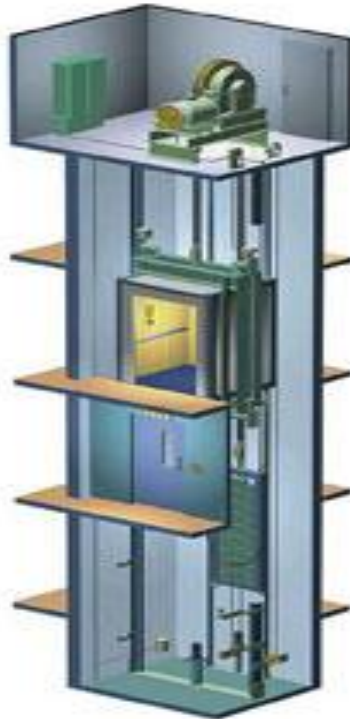
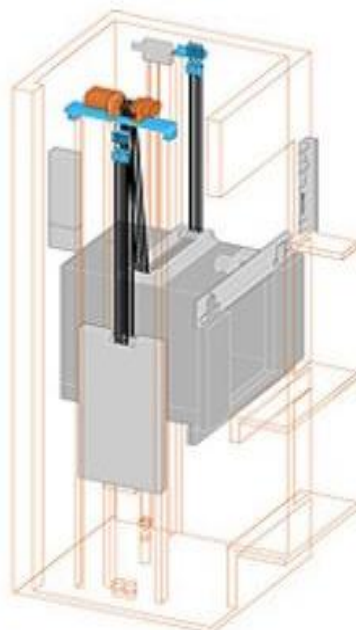


Figura 2.4 Ascensor Electromecánico

2.2.1.2 Ascensores autoportantes

Los ascensores autoportantes tienen la particularidad de que no necesitan sala de máquinas, por lo que permiten un ahorro en la instalación y su espacio puede ser ocupado para la construcción de una nueva planta del ascensor. Estos ascensores tienen un motor mucho más reducido, lo que les permite instalar el sistema de tracción en el mismo hueco del ascensor.

Este tipo de ascensores, además del comentado ahorro económico, tienen la particularidad de que pueden ser instalados casi en cualquier lugar, lo que los hace idóneos para ser los elegidos en edificios ya construidos que no poseen ascensor sin la necesidad de realizar cambios drásticos en la estética de los edificios. Además, son utilizados en viviendas unifamiliares y en estructuras donde sea necesario prescindir de sala de máquinas [9].



Fifura 2.5 Ascensor Autoportable

2.2.1.3 Ascensores hidráulicos

Estos ascensores poseen un motor eléctrico acoplado a una bomba que introduce aceite a presión en el cilindro, impulsando la cabina para provocar su ascenso. En la bajada, se deja vaciar el pistón mediante una válvula que permite el descenso del émbolo. La maquinaria y el depósito de aceite se pueden instalar hasta a 12 metros del hueco del ascensor, lo que permite que puedan ser instalados prácticamente donde se desee.

Estos ascensores tienen una velocidad más controlada que los electromecánicos por lo que los hace más cómodos además de ser más seguros, ya que carecen de contrapesos. Al no disponer de contrapesos, estos ascensores consumen mayor energía en el ascenso, pero este consumo de energía es compensada en el descenso por lo que, en general, son más económicos.

Como aspecto negativo, debido al sistema de elevación utilizado, estos ascensores son más lentos que los ascensores electromecánicos. Además, por temas de seguridad, no son aconsejables para pisos superiores a 21 metros de altura [10] [11].



Figura 2.6 Ascensor Hidráulico

2.2.2 Control de maniobra de los ascensores

Existen diferentes maniobras en función del sistema de gestión del tráfico que se haya instalado en el ascensor. Así, no será el mismo tipo de maniobra en un edificio de viviendas, donde los movimientos habituales de los usuarios son los de desplazarse hacia la planta baja o subir a alguna planta desde esta planta baja, que el de un edificio de oficinas de una empresa, donde el tráfico de personas discurre entre las distintas plantas del edificio.

Es por esto por lo que, en función del tipo de edificio donde vaya a ser instalado el ascensor, ha de serle implantado un tipo de maniobra específica.

2.2.2.1 Maniobra universal

Con este tipo de maniobra, se utiliza un solo botón para llamar al ascensor, independientemente de la dirección del viaje y de la parada. Estos ascensores registran la primera llamada que se produce y se atiende. Hasta que no termina el movimiento de la llamada realizada no se admite una nueva.

Las llamadas realizadas desde la cabina tienen preferencia, esto es, en el momento en que se cierra el ascensor se mantienen bloqueados los pulsadores de llamada exterior durante 2-4 segundos (dependiendo del fabricante) para dar la prioridad a los usuarios que se encuentren en el interior de la cabina. Pasado este tiempo sin realizarse

llamada alguna, la próxima petición del ascensor será la atendida, independientemente del lugar desde el cual haya sido solicitada.

Existe una variante de este tipo de maniobra, la botonera con memoria, la cual registra todas las llamadas realizadas pero se atenderán en el orden en que han sido almacenadas.

En el caso de que existan dos o tres ascensores, dúplex o triplex respectivamente, cada ascensor tendrá sus propias llamadas. En caso de que ambos ascensores estén vacíos, acudirá a la llamada el ascensor más cercano y, si uno de los dos está ocupado, en caso de producirse una nueva llamada, la atenderá aquel ascensor que se encuentre libre. Cuando todos los ascensores se encuentran atendiendo distintas peticiones de los usuarios, los botones exteriores de los pasillos quedarán bloqueados hasta que uno se quede libre, impidiendo realizar llamada alguna.

La aplicación de este método se resume a ascensores industriales o de edificios de pocas plantas en los cuales el tráfico de persona es escaso. El mayor problema de este tipo de maniobra es el gran gasto energético.

2.2.2.2 Maniobra colectiva en bajada

En estos ascensores, las llamadas se registran tanto desde el interior de la cabina como las realizadas desde el exterior de ella, en los botones instalados en cada piso, reconociendo el ascensor desde donde han sido realizadas estas llamadas.

Cuando el ascensor se encuentra en subida, la cabina solo atenderá las llamadas realizadas desde el interior, en orden lógico, no por prioridad, pasando por alto las que han sido efectuadas desde los pasillos. En el caso de que el ascensor se encuentre en bajada, el ascensor atenderá todas las llamadas, igualmente en orden lógico, efectuadas tanto desde el interior de la cabina como las de los botones de pasillo de cada planta. Así, se consigue que la inversión del sentido del ascensor sea mínimo, permitiendo un ahorro energético mucho mayor que en el caso anterior.

En caso de haber más de un ascensor, cuando ambos ascensores se encuentran vacíos durante un tiempo prolongado, uno de ellos se suele programar para que sea aparacado en la planta baja (planta de acceso) y el otro, en la última planta en la que ha realizado su parada. En caso de que dicha planta sea la planta de acceso, éste se situará en una planta intermedia.

Este sistema es utilizado en ascensores de edificios con escaso tráfico entre plantas, tales como clientes de hoteles, visitantes en hospitales o edificios de viviendas.

2.2.2.3 Maniobra colectiva selectiva en ambos sentidos de la marcha

Este sistema permite que los usuarios del ascensor no realicen movimientos innecesarios en el sentido inverso al de su marcha, a la vez que se alcanza el piso de destino con gran rapidez.

En los pasillos de cada planta, se pueden encontrar dos botones, uno para subir y bajar, excepto en las plantas extremo, que se utilizarán en función de la necesidad del usuario. Así, este tipo de ascensores atenderán todas las peticiones realizadas por el usuario, independientemente de si han sido realizadas desde el interior de la cabina o desde los pasillos, siempre y cuando la llamada realizada sea para realizar un movimiento en el mismo sentido que el realizado por la cabina en ese momento, es decir, si la cabina está subiendo, el ascensor atenderá, en orden lógico, exclusivamente las llamadas realizadas con el fin de ascender de planta. En cambio, si la cabina está bajando, tan solo se atenderán las peticiones para descender hacia plantas inferiores.

En este tipo de maniobra, en caso de que hayan sido instalados más de un ascensor, el sistema de aparcamiento de las cabinas serán igual que en la maniobra colectiva en bajada.

Esta maniobra es utilizada en edificios con un alto tráfico entre plantas, como pueden ser los ascensores para los empleados de un hotel u hospital, un edificio de oficinas o unos grandes almacenes [12] [13].

3 ASCENSOR ELECTROMECHANICO

En este capítulo se va a tratar con mayor profundidad los ascensores eléctricos con sala o cuarto de máquinas ya que la maqueta que se utilizará en este proyecto equivaldría a este tipo de ascensores. Con mayor detenimiento se analizarán cada una de las partes principales que componen estos ascensores.

Los ascensores eléctricos con sala de máquinas constan de tres partes principales en las cuales se agruparán los diferentes elementos que forman el ascensor en su conjunto. Estas partes son la propia sala de máquinas, el hueco del ascensor y el foso [14] [15].

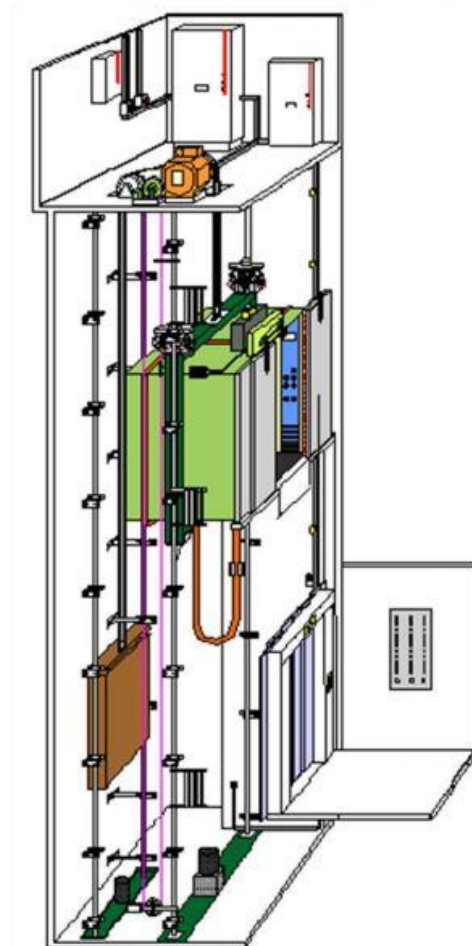


Figura 3.1 Esquema de un ascensor electromecánico con sala de máquinas

3.1 Sala de máquinas

Son locales especializados para la ubicación del grupo tractor, el limitador de velocidad y el cuadro de maniobra. Este cuarto de máquinas es una zona de acceso restringido, es decir, solo personal autorizado podrá acceder al interior de estos, estando prohibida la entrada para personas ajenas a este personal. Se debe instalar sobre el hueco del ascensor y nunca debe encontrarse adosado a locales habitados. Según la norma UNE EN 81-1, en caso de acceso mediante escalera, ésta debe tener una anchura mínima de 70 cm y una inclinación máxima de 60°. Además, la sala debe tener una altura mínima de 1,8 m desde el suelo y la temperatura en el interior debe encontrarse entre los 5 y los 40 °C.

3.1.1 Grupo tractor

El grupo tractor es el encargado de trasladar el movimiento de giro de un motor al desplazamiento de una polea para permitir el movimiento de la cabina y el contrapeso. Los componentes principales del grupo tractor son el motor, el freno, la reductora de velocidad, las poleas de tracción y el volante de inercia.



Figura 3.2 Grupo Tractor

3.1.1.1 Motor

El motor eléctrico es el encargado de transformar la energía eléctrica en energía mecánica producida por acción de los campos magnéticos que se generan gracias a las bobinas. El motor hace girar un eje con un tornillo sin fin que va conectado al reductor permitiéndole darle movimiento. En la actualidad, los motores de corriente alterna son los más utilizados, ya que los de corriente continua son más costosos de mantener.

Además, se pueden distinguir motores de una velocidad, los cuales hacen que la cabina tenga detenciones más bruscas cuando llega al piso deseado; motores de dos velocidades, que varían su velocidad haciéndola más suaves cuando está próxima a su punto de parada; y motores de variador de frecuencia que permiten un amplio rango de velocidades.

3.1.1.2 Freno

Es un sistema que, en caso de ausencia de corriente eléctrica o de una sobrecarga nominal de un 25%, se activa provocando la detención del ascensor. Una vez reinstaurada la corriente o eliminada esa sobrecarga, este freno se podrá desbloquear manualmente.

3.1.1.3 Reductor

Como su propio nombre indica, el reductor tiene como fin reducir la velocidad del motor. El reductor está formado por un tornillo sin fin de acero que engrana a una corona dentada de bronce, montados en el interior de una carcasa. Esto permite que la velocidad proporcionada por el motor y transmitida al tornillo se vea reducida al pasar por la corona dentada. Su función es permitir que el ascensor se pueda mover a una velocidad más baja que la proporcionada por el motor haciéndolo más seguro.

En los ascensores más novedosos, los cuales alcanzan velocidades superiores a los 2 m/s, se tiende a utilizar grupos tractores sin reductor. Este tipo de grupos son los denominados *gearless*.



Figura 3.3 Interior del reductor de un grupo tractor

3.1.1.4 Polea de tracción

Es la encargada de realizar el movimiento de subida y de bajada de las correas de la cabina y del contrapeso, gracias a la velocidad proporcionada por el reductor, en caso de tenerlo. Si no posee reductor, vendrá otorgada directamente por el motor. Las poleas vienen determinadas por tres características que las definen: su diámetro, el perfil de las gargantas y el material.

El diámetro de la polea vendrá determinado por la velocidad de desplazamiento del ascensor. Así, a mayores diámetros de la polea, mayor velocidad del ascensor pero también mayor será el esfuerzo del motor. El diámetro mínimo debe ser 40 veces el diámetro de los cables de suspensión.

El perfil de las gargantas de las poleas tendrá una influencia directa en la duración de los cables. Si la polea es demasiado estrecha el cable queda enclavado en ella; por contra, si es demasiado ancha, el cable se aplasta. Se pueden destacar los perfiles de garganta trapezoidales y semicirculares.

El perfil trapezoidal o de cuña tiene una gran adherencia entre la polea y los cables pero provoca un rápido desgaste de los cables y de la garganta. Por su parte, los perfiles semicirculares tienen una menor adherencia pero como punto positivo destaca una mayor duración de los cables y de la garganta.



Figura 3.4 Polea de tracción de un grupo tractor

3.1.1.5 Volante de inercia

El volante de inercia tiene como objetivo asegurar que el ascensor quede nivelado correctamente en cada piso. Es utilizado en ascensores cuyos motores son de una única velocidad para que la precisión de la llegada a la cabina sea mayor a la par que menos brusca.

3.1.2 Limitador de velocidad

El limitador de velocidad es un sistema de seguridad que entra en funcionamiento en el momento en que el ascensor adquiere una sobrevelocidad de un 25% por encima de la velocidad nominal. Está formado por dos poleas: una situada en el cuarto de máquinas (limitador de velocidad) y otra situada en el foso (polea tensora del limitador). Ambas trabajan solidariamente gracias a un cable que las conecta y que está anclado al bastidor de la cabina. Dicho cable se encuentra unido por uno de sus ramales al mecanismo de paracaídas de la cabina, el cual es otro sistema de seguridad.

En el momento en que el ascensor adquiere una sobrevelocidad superior a la permitida, la polea del limitador de velocidad se bloquea haciendo que el cable se tense y provoque un tirón en la palanca del paracaídas [16].



Figura 3.5 Limitador de velocidad

3.1.3 Cuadro de maniobra

Es el llamado “cerebro del ascensor”. Es el encargado de que el ascensor realice los movimientos de subida y de bajada cada vez que se produce una pulsación de llamada, es decir, ejecuta la maniobra que se le haya programado al ascensor en cuestión. En él, se encuentra el microprocesador o el PLC que contenga toda la información necesaria para la ejecución de los movimientos del ascensor en función de las llamadas, además de la inclusión de los diferentes relés y contactores necesarios para la correcta ejecución del motor del ascensor [17].



Figura 3.6 Cuadro de maniobras

3.2 Hueco del ascensor

Es el espacio vertical donde se alojan la(s) cabina(s) del ascensor. Además de estas, en el hueco del ascensor se encuentran otros muchos elementos que componen el conjunto del ascensor. Estos elementos son las guías y fijaciones de la cabina, el contrapeso, las guías y fijación del contrapeso, los cables, el paracaídas y los diferentes sensores que mantendrán al cuadro de maniobra en conocimiento en todo momento del estado y localización del ascensor.

3.2.1 Cabina

Es el elemento portante del ascensor. Las cabinas están formadas principalmente por dos elementos: el bastidor y la caja que forma la propia cabina [18].



Figura 3.7 Cabina de ascensor

3.2.1.1 Bastidor

El bastidor es un elemento de metal cuya función principal es la de sostener la caja que conforma la cabina del ascensor. Para ello, debe ser robusto, resistente y cumplir unos requisitos mínimos de seguridad. Además de sostener a la cabina, a él se fijan los cables de la suspensión de la cabina y el sistema de seguridad del paracaídas.

3.2.1.2 Caja

Es el elemento portante en sí, el habitáculo al cual las personas acceden para realizar los desplazamientos entre plantas. Según la Norma UNE EN-81, la caja debe estar completamente cerrada por las paredes, el suelo y el techo debiendo ser de materiales metálicos. La cabina también incluye las puertas que darán acceso a la misma, que según se recogen en la Norma, deben tener una altura de 2 metros. Además, en función del número de personas y peso máximo admisible, las dimensiones de la cabina serán de tamaños diferentes.

3.2.2 Guías y fijación de la cabina

Las guías son las encargadas de conducir a la cabina durante los trayectos de subida y de bajada del ascensor y sirven de apoyo en caso de rotura de los cables, impidiendo la caída súbita del ascensor y su desnivelación. Debido a esto, las guías del ascensor deberán tener una resistencia acorde al peso que deberán soportar, además de ser revisadas periódicamente.

Las guías más utilizadas en los ascensores actuales poseen una forma de T, aunque también es posible encontrarlas con forma de V invertida.

3.2.3 Contrapeso

El contrapeso es un elemento que se utiliza para equilibrar las cargas y es esencial a la hora de ascender y descender a una planta. Su masa es igual a la de la cabina más la mitad de la carga máxima autorizada en ella, lo que permite que el motor no tenga la necesidad de mover toda la masa cabina y, por tanto, no trabaje a máximo rendimiento ni aun con la cabina ocupada con su carga máxima.

Sin la existencia del contrapeso la posibilidad de accidente sería elevada, además de producirse inestabilidades en la carga, desajustes de velocidad e inseguridad [19].

El contrapeso está compuesto por unas pesas que conforman la mayor parte de la masa, los cables de suspensión de donde cuelgan los cables de la polea, los marcos de los perfiles que propicia la sujeción de las pesas y las patas, que mantiene al contrapeso en todo momento en contacto con las guías.

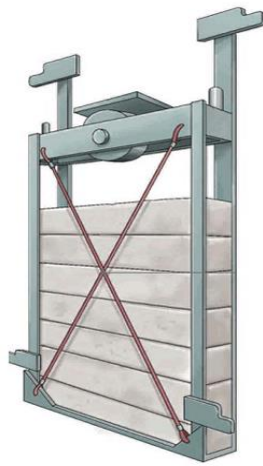


Figura 3.8 Contrapeso de un ascensor

3.2.4 Guías y fijaciones del contrapeso

Al igual que la cabina, el contrapeso también precisa de la utilización de guías que permita el movimiento del mismo por el carril sin producir desviaciones. También, actúan como protección en caso de emergencia ante la posible rotura de cables, ya que ayudan al soporte del contrapeso con mayor facilidad.

3.2.5 Cables

Son los encargados de la sujeción de la cabina y del contrapeso recorriendo todo el hueco del ascensor y la polea del grupo tractor. El material utilizado por estos cables es el acero, los cuales están constituidos de alambres agrupados formando cordones, que a su vez se enrollan en un alma formando finalmente el conjunto apto para la resistencia de los diferentes esfuerzos.

Según el Real Decreto 203/2016, en el que se establecen los requisitos y componentes esenciales de seguridad de los ascensores, éstos deben tener un mínimo de dos cables que realicen la sujeción tanto de la cabina como del contrapeso, con su respectivo sistema de enganche [20].

3.2.6 Paracaídas

Es un sistema de seguridad situado en el bastidor de la cabina que provoca el frenado del ascensor cuando se produce una caída de la corriente eléctrica o un corte en los cables de sustentación. En el momento en el que el limitador de velocidad actúa debido al corte en el suministro de la corriente, se produce un tirón en el cable que sostiene a la cabina por lo que, a su vez, también tira del paracaídas. En el momento en el que actúa el paracaídas se acciona una timonería¹ que hace desplazar en dirección vertical unas varillas de actuación.

En el mercado se pueden encontrar dos tipos de paracaídas:

- Paracaídas de acción instantánea: en ellos el limitador tira de la timonería accionando las zapatas², las cuales agarran cada vez con más fuerzas las guías de la cabina. Este sistema está poco aconsejado para ascensores en los que la velocidad es elevada ya que produce un frenado en seco demasiado brusco respecto a la velocidad que el ascensor tiene.
- Paracaídas de acción progresiva: actúan de la misma forma que los de acción instantánea pero frenan la caída aplicando sobre las zapatas de freno una fuerza de magnitud controlada por lo que no se produce una parada tan brusca [21].

3.3 Foso

El foso es el recinto inferior del hueco del ascensor pero lo que destaca de éste es que está situado por debajo de la planta más baja del ascensor, por lo que la cabina nunca podría acceder hasta él. Al igual que debe ocurrir con el hueco del ascensor, el foso debe estar libre de tuberías de agua o gas e impermeabilizados, ya que en numerosas ocasiones están situados por debajo del nivel del suelo y es posible que se produzcan filtraciones.

El foso debe ser lo suficientemente profundo como para que, en caso de que la cabina o el contrapeso compriman en su totalidad sus amortiguadores, siga quedando un espacio nunca inferior a 0.5 m. En el foso se ubicarán los topes o amortiguadores de la cabina y del contrapeso y una serie de interruptores de seguridad.

Para poder realizar el acceso al foso, es obligatorio tener instalada una escalera que permita su entrada.

3.3.1 Amortiguadores

Los amortiguadores se encargan de frenar la llegada de la cabina o del contrapeso según sean amortiguadores de cabina o de contrapeso, respectivamente, convirtiendo la energía cinética recibida por el impacto en energía térmica disipada por el fluido que contienen. Es un sistema de frenado que sirven tanto para frenar la llegada al piso más bajo evitando una parada demasiado brusca, como en caso de emergencia si se produce una rotura de los cables, que impida que la cabina o el contrapeso golpeen con violencia sobre la superficie.

Cabe la posibilidad de que estos amortiguadores sean colocados en el bastidor de la cabina y del contrapeso en lugar de ser colocados en el foso, aunque esta opción es menos habitual.

Los amortiguadores están formados por un muelle encargado de recibir el impacto, y el propio amortiguador, conformado por un tubo de reserva, un tubo de presión una válvula, un pistón y la cabeza del pistón. Por el interior del conjunto circula aceite, que con el movimiento del muelle y el amortiguador, provocará que la energía cinética adquirida por el muelle en el impacto se convierta en energía calorífica.

Se distinguen principalmente tres tipos de amortiguadores en función de la velocidad que tenga el ascensor:

- Elásticos: son amortiguadores de caucho que se utilizan para los ascensores más antiguos y, a su vez, más lentos, aquellos que no superen los 60 cm/s.
- De resorte: son utilizados en ascensores donde la velocidad de la cabina no supere los 1.75 m/s.
- Hidráulicos: se pueden utilizar en cualquier caso.

¹ Conjunto de tirantes y palancas que amplifican y transmiten los esfuerzos de los cilindros de freno, volantes de freno y demás actuadores en los sistemas de freno, a los elementos de apriete.

² 3. f. Pieza del freno que actúa por fricción sobre el eje o contra las ruedas para moderar o impedir su movimiento.



Figura 3.9 Amortiguador del foso de un ascensor

3.3.2 Interruptores de seguridad

Son aquellos interruptores que sirven para detener el movimiento del ascensor y mantenerlo inmovilizado de manera manual. Se utilizan para situaciones de reparación, mantenimiento y otras acciones de la misma índole.

4 MAQUETA DE ASCENSORES Y AUTÓMATA

Una vez conocidos los elementos que componen un ascensor eléctrico con sala de máquinas y como funcionan cada uno de ellos, se procederá a la explicación de la maqueta de la que disponemos y del autómata programable, o más comúnmente conocido en el ámbito de la automatización por sus siglas en inglés PLC.

Además de esto, se explicará con detenimiento el sistema de conexionado que se ha llevado a cabo para conseguir modernizar el sistema de control y poder ser ejecutado por el autómata.

4.1 Maqueta del ascensor

La maqueta consta de dos sistemas de ascensores de 4 plantas cuya “sala de máquinas” está situada en la parte superior, ya que es allí donde se encuentran localizados los motores de los ascensores.

La maqueta tiene unas dimensiones de 75 cm de alto, 30 cm de ancho y 17 cm de profundidad cuyo material utilizado para el armazón es el alumnio. Las cabinas, de acero inoxidable, tienen unas medidas de 7 cm de ancho, 7 cm de largo y 10 cm de alto. Para la simulación de las personas se dispone de un bloque de hierro con forma cilíndrica de aproximadamente 400 g de masa. Para el contrapeso de cada cabina se dispone de un bloque del mismo material pero de mayor tamaño, por lo que su masa es mayor, 450 gramos. El cable encargado de unir la cabina con el contrapeso es un hilo de sedal lo suficientemente resistente como para soportar el peso tanto de la cabina como del contrapeso sin romperse.

Para la simulación de las luces de los pulsadores de pasillo, hay instalados 6 leds rojos que formarán parte de las salidas del automata. Estos 6 leds equivalen a las luces de llamada de pasillo del primer piso para subir, del último piso para bajar, y de los 4 leds restantes, dos serán para la elección de subir de las 2 plantas intermedias y los otros dos para la elección de bajar. Además de estos 6 leds, se dispone de un total de 18 pulsadores, que serán entradas para el autómata y que tendrán su equivalente en: los 6 pulsadores exteriores de pasillo, al igual que los leds; y 6 pulsadores interiores por cabina. De estos 6 pulsadores, 4 serán utilizados para la elección de cada una de las 4 plantas y los otros dos se utilizarán para permitir la parada y reactivación del ascensor simulando, por ejemplo, una parada de emergencia y su posterior restauración.

Además de estas 18 entradas, hay que añadir 8 entradas que serán las originadas por los sensores de planta, los cuales son contactores situados a la izquierda de cada cabina al nivel de cada piso. Al llegar a ellos y ser del mismo material que las cabinas, hacen contacto y permiten la activación de la entrada correspondiente.

Por último, existen 4 salidas más que corresponden a los motores y que permitirán el movimiento de la cabina y del contrapeso. Los motores son de corriente continua de 12 VCC y unos 3 A por lo que la potencia máxima consumida es de 72 W.

Cada motor, al entrar en funcionamiento al recibir corriente eléctrica, hacen girar un eje. Este eje a su vez hace girar una rueda dentada acorde a él. Esta rueda está atravesada por otro eje de hierro al cual hace rotar al compás de su movimiento. Finalmente, en este eje se encuentra enrollado el hilo hasta en 3 ocasiones para impedir su soltura, que sujeta tanto a la cabina como al contrapeso y que permite moverlos gracias a estar guiado por una pequeña polea.

Para evitar que los cables tanto de las entradas como de las salidas se encuentren desordenados e impidan el correcto funcionamiento de la maqueta, se recogen por los laterales de la maqueta de una forma ordenada y limpia, culminando todos agrupados en dos conectores DB 25 macho. La relación de los cables con cada uno de los pines de los DB se recogen en las siguientes tablas.

CONECTOR	CABLE EQUIVALENTE
PIN 1	Botón 0 interior de la cabina derecha
PIN 2	Botón 1 interior de la cabina derecha
PIN 3	Botón 2 interior de la cabina derecha
PIN 4	Botón 3 interior de la cabina derecha
PIN 5	Botón S (Stop ascensor) interior de la cabina derecha
PIN 6	Botón A (Activar ascensor) interior de la cabina derecha
PIN 7	Libre
PIN 8	Libre
PIN 9	Botón 0 interior de la cabina izquierda
PIN 10	Botón 1 interior de la cabina izquierda
PIN 11	Botón 2 interior de la cabina izquierda
PIN 12	Botón 3 interior de la cabina izquierda
PIN 13	Botón S (Stop ascensor) interior de la cabina izquierda
PIN 14	Botón A (Activar ascensor) interior de la cabina izquierda
PIN 15	Libre
PIN 16	Libre
PIN 17	Botón 0 pasillo subida
PIN 18	Botón 1 pasillo subida
PIN 19	Botón 2 pasillo subida
PIN 20	Botón 1 pasillo bajada
PIN 21	Botón 2 pasillo bajada
PIN 22	Botón 3 pasillo bajada
PIN 23	Libre
PIN 24	Libre
PIN 25	Libre

Tabla 4.1 Correlación cables-conectores del 1º DB 25

CONECTOR	CABLE EQUIVALENTE
PIN 1	Común de los leds
PIN 2	Motor 1 (Subida)
PIN 3	Motor 1 (Bajada)
PIN 4	Libre
PIN 5	Sensor planta 0 cabina izquierda
PIN 6	Sensor planta 1 cabina izquierda
PIN 7	Sensor planta 2 cabina izquierda
PIN 8	Sensor planta 3 cabina izquierda
PIN 9	Sensor planta 0 cabina derecha
PIN 10	Sensor planta 1 cabina derecha
PIN 11	Sensor planta 2 cabina derecha
PIN 12	Sensor planta 3 cabina derecha

PIN 13	Libre
PIN 14	Común pulsadores
PIN 15	Motor 2 (Subida)
PIN 16	Motor 2 (Bajada)
PIN 17	Led planta 0 pasillo subida
PIN 18	Led planta 1 pasillo subida
PIN 19	Led planta 2 pasillo subida
PIN 20	Led planta 1 pasillo bajada
PIN 21	Led planta 2 pasillo bajada
PIN 22	Led planta 3 pasillo bajada
PIN 23	Libre
PIN 24	Libre
PIN 25	Libre

Tabla 4.2 Correlación cables-conectores del 2º DB 25

Así pues, una imagen de la maqueta de la que se dispone para este trabajo es la siguiente.



Figura 4.1 Maqueta ascensor utilizada en el proyecto

4.2 Autómata programable

Un autómata programable es un equipo electrónico, programable en lenguaje no informático, diseñado para controlar en tiempo real y en ambiente industrial, procesos secuenciales. Los PLCs trabajan en base a la información recibida por los captadores y actuando según el programa lógico interno que contenga.

Sus campos de aplicación son muy variados debido a su reducido tamaño y a la facilidad de programación que tienen, destacando su utilización en procesos de maniobras de máquinas, maniobras de instalaciones y en acciones de control, seguridad y señalización. Su función será la de manejar la maqueta de ascensores disponible, aunque también podría utilizarse para el control de un sistema real de ascensores.

El PLC utilizado será el Modicon M340 de Schneider Electric. Los procesadores de esta autómata gestionan toda la estación PLC, formada por un módulo de E/S binarias, módulos de E/S analógicas y módulos de conteo, otros módulos expertos y módulos de comunicación. Éstos se distribuyen en uno o más bastidores conectados al bus local. Cada bastidor debe incluir un módulo de alimentación. El bastidor principal alojará la CPU.

Sin embargo, para el control de la maqueta, simplemente será necesario, como se muestra en la imagen siguiente de izquierda a derecha: un módulo de alimentación (BMX CPS 2000), un procesador P34 2020, un módulo de Entradas binarias (BMX DDI 3202K), un módulo de E/S binarias (BMX DDM 16002). Todo irá situado en un bastidor principal de 4 ranuras (BMX XBP 0400).

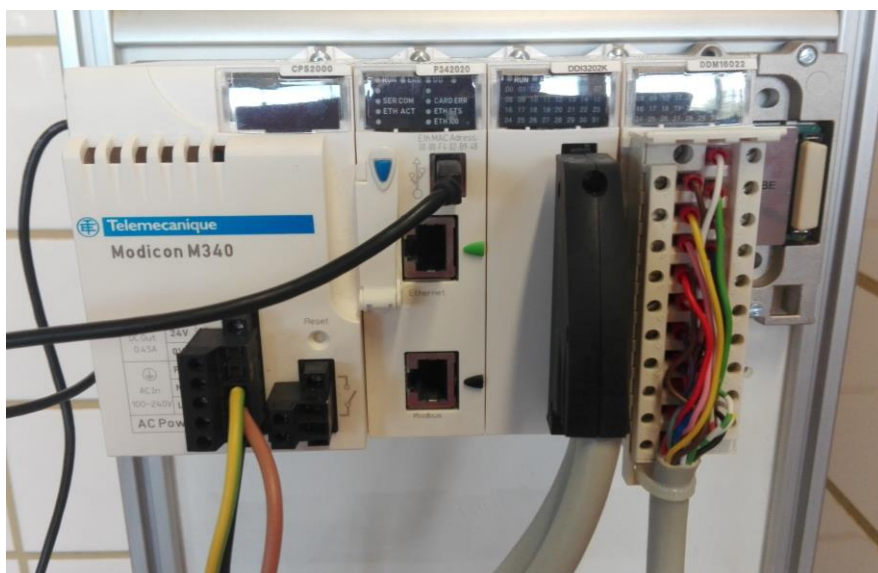


Figura 4.2 Autómata programable Modicon M340

4.2.1 Bastidor BMX XBP 0400

Constituye el elemento fundamental de la estación del PLC M340 de Modicon. Éste desarrolla una función mecánica, ya que se encarga de la fijación de todos los módulos de la estación del PLC. Aunque lo normal es ser instalado en un rail DIN, ha sido instalado de forma atornillada a un panel. Además, también realiza una función eléctrica ya que el bastidor proporciona la alimentación necesaria para cada módulo en un solo bastidor.

Este bastidor cuenta con 4 ranuras, las cuales irán numeradas de 00 a 03. Además incluye dos ranuras más: una al principio que será exclusivamente para instalarse el módulo de alimentación y viene marcada como CPS, y otra al final del bastidor, reservada para un módulo de expansión, que nos permitirá conectar el bastidor con otro para ampliar el número de módulos, marcada con las letras XBE. La ranura 00 está reservada para la instalación del procesador, mientras que el resto de ranuras será para la instalación de módulos de E/S. Hay que destacar, que en el caso de este proyecto tanto la ranura de expansión como la ranura 03 no serán necesarias y, por tanto, no se utilizan.

4.2.2 Módulo de alimentación BMX CPS 2000

Debido a que el autómatas se alimenta de corriente alterna procedente de la red eléctrica, se ha utilizado el módulo BMX CPS 2000 ya que es una de las opciones disponibles cuando la corriente es en alterna. Este módulo posee los siguientes componentes:

- Un panel de visualización, el cual contiene dos elementos: un indicador LED OK (verde) encendido si el módulo de la fuente de alimentación del bastidor está operativa y funciona correctamente; y un indicador LED de 24 VCC (verde) encendido si la fuente de alimentación del sensor está en funcionamiento.
- Un relé de alarma, que en modo normal, se activa y se mantiene cerrado su contacto, pero en el momento en que se produce una detención, el relé se cierra y su contacto asociado se abre. Esta detención se puede deber a la aparición de un error con bloqueo, a una tensión de salida incorrecta de la fuente de alimentación del bastidor o a la desaparición de la tensión de red. Dicho relé no se utilizará en el proyecto.
- Un botón de RESET que, al pulsarse, provoca un corte en la alimentación; y al soltarse, produce un nuevo arranque, es decir, se interpreta estas acciones como un reinicio en frío.
- Un conector para la red de entrada, que será por donde se le suministre la corriente alterna procedente de la red eléctrica. Dentro del módulo se encuentra un fusible que sirve de protección conectado a la fase de entrada de la red de corriente alterna. Este módulo contiene además una fuente de alimentación interna de 24 VCC con sus respectivos sensores exteriores para ser conectados. Esta fuente tiene una disponibilidad de 0.45 mA.

Al igual que el relé de alarma, la fuente de alimentación de 24 VCC tampoco será utilizada, ya que se dispone de una fuente de alimentación externa que proporciona mayor potencia y que permite un conexionado más limpio.

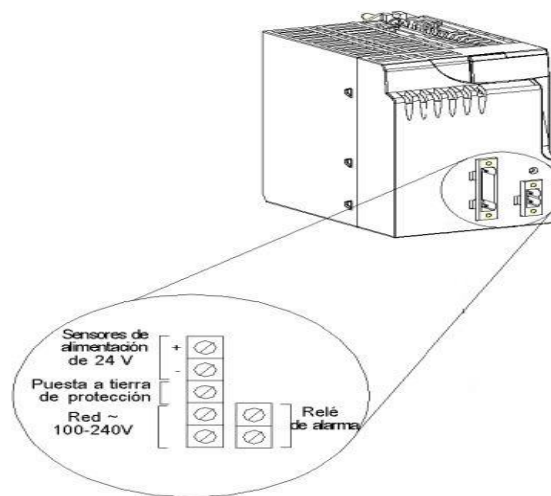


Figura 4.3 Esquema de la alimentación y puesta a tierra del BMX CPS 2000

4.2.3 Procesador BMX XBP 0400

Es el encargado de gestionar toda la estación del PLC, es decir, los módulos de E/S binarios, los módulos de E/S analógicos y los módulos de comunicación en caso de que los haya.

El procesador BMX P34 2000 permite un máximo de 1024 E/S binarias, 256 E/S analógicas y un tamaño de memoria de hasta 4096 Kb. Está compuesto de un panel de visualización que, mediante indicadores leds, permiten comprobar el estado del procesador. También, contiene un puerto USB, un puerto serie y un puerto Ethernet, dando así la libertad de poder utilizar el medio de comunicación entre el autómatas y el programa que más convenga al usuario. En el caso de este proyecto, la comunicación se realizará mediante el puerto USB.

Por último, el procesador contiene una tarjeta de memoria con su correspondiente protección. Esta tarjeta de memoria es totalmente necesaria para el funcionamiento del autómatas y está formateada para utilizarse con productos de Schneider Electric, por lo que no debe formatearse ni manipularse en otro entorno [22].

4.2.4 Módulo BMX DDI 3202K

Es un módulo de entradas binarias que utiliza una tensión nominal de entrada de 24 VCC y una corriente de 2.5 mA que va conectado a través de un conector de 40 pines. De esos 40 pines, 32 son pines de entradas alimentados por la corriente procedente de los sensores. La lógica que utiliza este tipo de módulos es una lógica positiva o común positivo, es decir, las señales de tensión alta son representadas con un 1 binario (una tensión superior a 11 V y una corriente mayor a 2 mA), mientras que las señales de tensión baja se representan con un 0 binario (una tensión de 5 V y una corriente inferior a 1.5 mA).

En este módulo, una vez conectado, cuando se produce el cierre de uno de los contactos, se produce la llegada de la tensión y por tanto, el módulo reconoce la activación de la entrada correspondiente a ese contacto. Al producirse esta activación, dicha entrada pasa a tener un 1 binario y se encenderá su luz correspondiente del panel de visualización, mientras que si el contacto se encuentra abierto o sin la suficiente tensión, la entrada se encontrará en el estado de 0 binario y su luz estará apagada.

Este módulo contará con dos mangueras de 20 cables cada una, con cables de diferentes colores que serán los encargados de indicar que cable corresponde a cada sensor del módulo. Las mangueras se encuentran unidas por lo que se debe aclarar que la manguera delantera corresponde a los 20 primeros sensores y la trasera, a los 20 últimos.

La leyenda de colores de la manguera 1 se presentan en la tabla que se muestra a continuación, además del diagrama de conexión del módulo a los sensores, en el que los fusibles utilizados son de 0.5 mA.

* Los colores de la manguera 2 son los mismos que los de la 1 y siguen el mismo orden de sensores, siendo los 20 sensores siguientes del módulo.

CANAL	COLOR
Sensor 1 (I0)	Blanco
Sensor 2 (I1)	Marrón
Sensor 3 (I2)	Verde
Sensor 4 (I3)	Amarillo
Sensor 5 (I4)	Gris
Sensor 6 (I5)	Rosa
Sensor 7 (I6)	Azul
Sensor 8 (I7)	Rojo
Sensor 9 (I8)	Negro
Sensor 10 (I9)	Morado
Sensor 11 (I10)	Gris-Rosa
Sensor 12 (I11)	Rojo-Azul
Sensor 13 (I12)	Blanco-Verde
Sensor 14 (I13)	Marrón-Verde
Sensor 15 (I14)	Blanco-Amarillo
Sensor 16 (I15)	Amarillo-Marrón
Sensor 17 (+24 V, SPS1+)	Blanco-Gris
Sensor 18 (-24 V, SPS1-)	Marrón-Gris
Sensor 19 (+24 V, SPS1+)	Blanco-Rosa
Sensor 20 (-24 V, SPS1-)	Marrón-Rosa

Tabla 4.3 Relación sensores-colores del BMX DDI 3202K

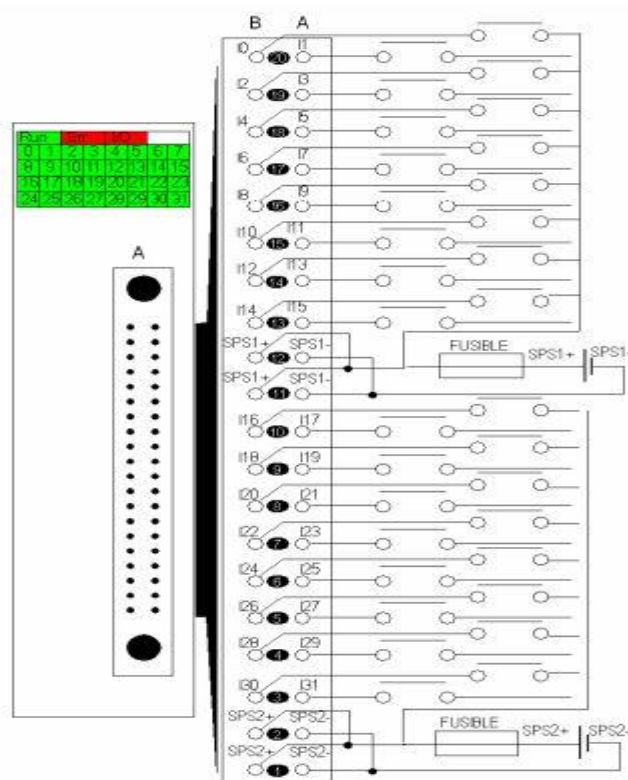


Figura 4.4 Diagrama de conexión del módulo BMX DDI 3202K

4.2.5 Módulo BMX DDM 16022

El módulo BMX DDM 16022 es un módulo binario de 24 VCC de tensión nominal y una corriente de 3.5 mA conectado a través de un bloque de terminales de 20 pines. Es un módulo de lógica positiva donde los 8 canales de entrada reciben corriente de los sensores (común positivo) y los 8 canales de salida proporcionan corriente a los preactuadores (común negativo).

En este caso, el 1 binario se consigue cuando la tensión supera los 11 V y la corriente alcanza al menos los 3 mA, mientras que el 0 binario se produce con una tensión de 5 V y una corriente inferior a los 1.5 mA.

Al igual que ocurría con el módulo BMX DDI 3202K, en este módulo las entradas y las salidas tienen el mismo funcionamiento: cuando el contacto de la entrada o el preactuador de la salida estén abiertos se mantendrán en 0 binario y la luz pertenecientes a la entrada o salida en cuestión estará apagada, mientras que si se cierran y circula la tensión suficiente por ellos, el estado de la entrada o salida será de 1 binario y se encenderá el led correspondiente del panel de visualización.

Este módulo cuenta con una manguera de 20 colores, los cuales son los mismos que los de las mangueras del módulo de entradas, pero su relación con los sensores es diferentes. Así, su leyenda de colores y su diagrama de conexión son los siguientes:

CANAL	COLOR
Sensor 1 (I0)	Blanco
Sensor 2 (I1)	Marrón
Sensor 3 (I2)	Verde
Sensor 4 (I3)	Amarillo
Sensor 5 (I4)	Gris
Sensor 6 (I5)	Rosa
Sensor 7 (I6)	Azul
Sensor 8 (I7)	Rojo
Sensor 9 (-24 V)	Marrón-Gris
Sensor 10 (+24 V)	Blanco-Gris
Sensor 11 (Q16)	Negro
Sensor 12 (Q17)	Morado
Sensor 13 (Q18)	Gris-Rosa
Sensor 14 (Q19)	Rojo-Azul
Sensor 15 (Q20)	Blanco-Verde
Sensor 16 (Q21)	Marrón-Verde
Sensor 17 (Q22)	Blanco-Amarillo
Sensor 18 (Q23)	Amarillo-Marrón
Sensor 19 (-24 V)	Marrón-Rosa
Sensor 20 (+24 V)	Blanco-Rosa

Tabla 4.4 Relación sensores-colores del BMX DDM 16022

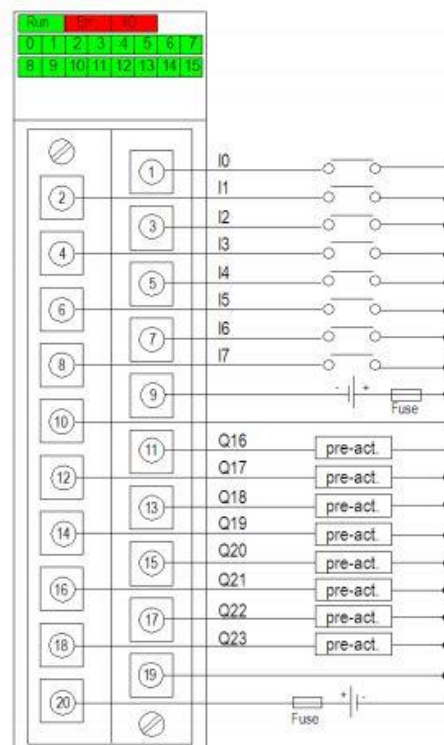


Figura 4.5 Diagrama de conexión del módulo BMX DDM 16022

En este módulo, el fusible de entrada no tiene el mismo amperaje que el fusible de salida. El fusible de entrada vuelve a ser de 0.5 mA, mientras que el fusible de salida se recomienda que sea de 6.3 mA. Para el conexionado de la maqueta el fusible de salida utilizada es menor, de 2 mA, aunque es más que suficiente para las corrientes que son manejadas [23].

4.3 Otros elementos

Una vez conocidos tanto el PLC que se va a utilizar como la maqueta de la que se dispone, lo siguiente a destacar son los elementos intermediarios que permitirán el correcto funcionamiento de la maqueta a través del autómat.

Estos elementos son los siguientes:

- 1 Fuente de alimentación TPC 055-124 de 24 VCC y 2.3 A.
- 1 Fuente de alimentación TPC 120-112 de 12 VCC y 8 A.
- 8 Relés de conmutación doble.
- 2 Placas IFM DEE 25 F de M-Jay Electronics ltd.
- 4 Fusibles (3 de 0.5 mA y 1 de 2 mA).
- 4 Resistencias de 2 K Ω .
- 3 Mangueras de 20 cables cada una.

En primer lugar, se disponen de los dos conectores DB 25 hembra a los cuales se conectarán los DB 25 M que se tiene de la maqueta, en concreto, los dos IFM DEE 25 que están preparados para ser instalados en un rail DIN y cuyo módulo dispone del propio conector hembra y terminales de tornillo del que podrán partir nuevos cables para la conexión con el autómat. Para conocer el pin del conector que corresponde con cada terminal, ambos se encuentran serigrafiados con un número que los relaciona.

Para la alimentación de los dos motores, se ha utilizado una fuente de alimentación externa de 12 VCC, que será la encargada de ofrecer la tensión necesaria para que en el momento en que sea necesario, los motores puedan ponerse en funcionamiento.

Además de la fuente de alimentación, los motores precisan de la utilización 4 de relés electromecánicos simples (para este proyecto se utilizan relés dobles pero haciéndolos funcionar como simples), dos por cada motor. Estos relés serán los encargados de permitir el movimiento del motor cuando se estipule conveniente. Así, con la utilización de los relés, se impide que se le pueda dar a un motor la orden de girar en ambas direcciones a la vez y quemar parte del circuito.

El funcionamiento y esquema de los relés con el motor es el siguiente:

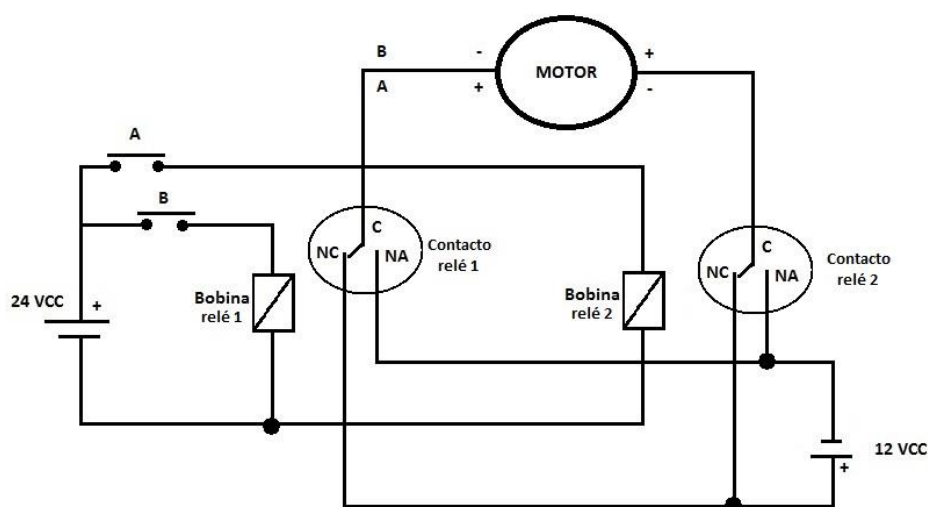


Figura 4.6 Esquema de la conexión del motor con los relés

Los interruptores A y B se traducen en el autómatas a una salida a 0 o 1 lógico que corresponderán a un estado de abierto o cerrado, respectivamente.

Si ningún interruptor está presionado, los motores, conectados a los comunes (C), mantendrán un circuito cerrado sin ningún tipo de alimentación al estar conectados a los pines Normalmente Cerrado (NC), por lo que no se moverán.

Si se presiona el interruptor A, el segundo relé salta a Normalmente Abierto (NA), formándose así un circuito cerrado que tiene una alimentación de 12 VCC, habiendo a la izquierda del motor un más (+) y a la derecha del motor un menos (-) por lo que el motor comienza a girar en una dirección. Si por el contrario el interruptor pulsado es el B, será el primer relé el que salte hacia NA y el motor se alimentará inversamente, por lo que el motor girará en la dirección contraria.

Por último, si se presionaran los dos interruptores a la vez, ambos relés estarían conectados a NA y se formaría un circuito cerrado sin ningún tipo de alimentación por lo que se volvería a mantener parado el motor. Así, se impide que se pueda activar a la vez el motor en ambos sentidos.

Por su parte, las bobinas de ambos relés, al ser de 24 VCC, deberán ir conectadas a la fuente de alimentación de 24 VCC.

Además de los relés de los motores, se han utilizado 4 relés para el encendido y apagado de los leds de las llamadas realizadas desde el exterior de las cabinas de la maqueta. Para ello, los leds han sido conectados a la fuente de 24 VCC a los que se les ha añadido una resistencia de 2 K Ω para impedir que la corriente que circule por los leds sea lo suficientemente alta como para quemarlos. Así, en un principio el relé se mantendrá en el estado NC, el cual estará abierto y, por tanto, no realizará ninguna acción. En cuanto se produzca un 1 en la salida de la bobina del relé, el relé saltará al estado NA que tendrá conectado la resistencia y el led a la fuente de 24 VCC, cerrando así el circuito y permitiendo que dicho led se encienda.

El esquema del led conectado al relé quedaría de la siguiente manera:

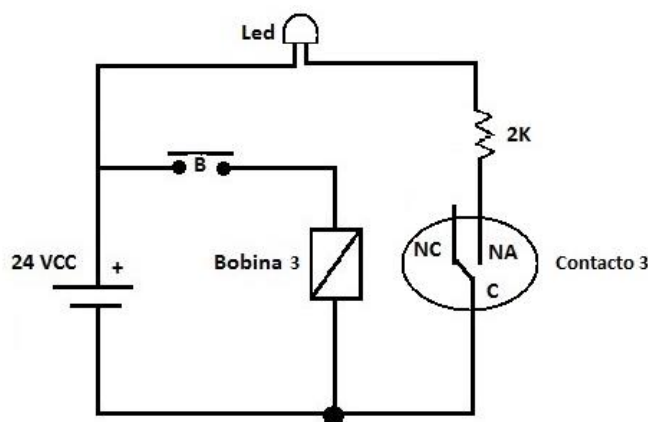


Figura 4.7 Esquema de la conexión de los leds con los relés

Los relés estarán situados en una placa que, al igual que ocurriera con los IFM DEE 25 F, contiene además de los conectores del relé, unos terminales de tornillos por los que conectar los diferentes cables. Las bobinas tienen unos terminales separados de los de las patas del relé. La nomenclatura se produce mediante letras y números, donde los relés están nombrados desde las letras I a la letra P. Así, para la bobina del relé, el terminal positivo corresponde a la letra del lugar donde se haya instalado el relé, mientras que el terminal negativo es común para todos los relés que se instalen, que para este proyecto será el máximo disponible (ocho), y tiene su correspondencia en los terminales con una línea (--). Por su parte, al ser relés de conmutación doble, se tienen 6 terminales por cada relé, teniendo 2 comunes, 2 normalmente abiertos y dos normalmente cerrados, aunque en la práctica tan solo se utilizará uno de cada, pero siempre pertenecientes al mismo conmutador, es decir, entre los números 1-3 o 4-6 sino no funcionará ya que estarían conectados a conmutadores diferentes. El común del relé corresponde a los números 1 y 4, el estado normalmente abierto corresponde a los números 2 y 5, y los números 3 y 6 corresponden al estado normalmente cerrado. Los relés disponen de sus propios fusibles para impedir sobretensiones.

Por último, se utiliza una fuente de alimentación externa de 24 VCC en lugar de la fuente que se tiene del autómat, ya que esta permite una conexión de cables más amplia y limpia al poseer dos terminales de conexión.

4.4 Conexionado

En este apartado, se dispone a explicar el circuito de conexionado entre el autómat, la maqueta y el resto de elementos.

Lo primero que se realiza es conectar tanto las fuentes de alimentación de 24 VCC y de 12 VCC como el módulo de alimentación BMX CPS 2000 con los cables correspondientes de la red eléctrica, conectando el cable verde a la tierra, el azul al neutro y el marrón a la línea. En la instalación, al solo contar con una clavija eléctrica, se han conectado sus cables a los de la fuente de alimentación de 24 VDC, realizando un puente entre las demás fuentes, por lo que los colores serán diferentes: en la fuente de alimentación de 12 VDC, la tierra será amarilla y verde, el neutro seguirá siendo azul y la línea, en este caso, será negro. Por su parte, el módulo de alimentación utilizará la tierra amarilla y verde, el neutro marrón y la línea, negra.

Una vez realizado el cableado de la red eléctrica, se procede a la conexión de ambos DB 25 macho de los cables de la maqueta a los dos DB 25 hembra de los IFM DEE 25 F. En la instalación, el IFM DEE 25 F instalado en la parte inferior del carril DIN, será el que contenga todas las entradas de las botoneras, en adelante, 1º DB; mientras que el IFM DEE 25 F instalado sobre el anterior, será el DB 25 que contiene tanto las entradas de los sensores como todas las salidas, en adelante, 2º DB.

Ya conectados, se realiza el cableado que irá desde los terminales de tornillos hasta los módulos de E/S del autómat.

Para ello, se realizan cableados directos de las entradas del 1º DB al autómat:

TERMINAL	COLOR	MÓDULO	MANGUERA	CANAL
Terminal 1	Blanco	DDI 3202K	1	I0
Terminal 2	Marrón	DDI 3202K	1	I1
Terminal 3	Verde	DDI 3202K	1	I2
Terminal 4	Amarillo	DDI 3202K	1	I3
Terminal 5	Gris	DDI 3202K	1	I4
Terminal 6	Rosa	DDI 3202K	1	I5
Terminal 9	Azul	DDI 3202K	1	I6
Terminal 10	Rojo	DDI 3202K	1	I7
Terminal 11	Negro	DDI 3202K	1	I8
Terminal 12	Morado	DDI 3202K	1	I9
Terminal 13	Gris-Rosa	DDI 3202K	1	I10
Terminal 14	Rojo-Azul	DDI 3202K	1	I11
Terminal 17	Blanco-Verde	DDI 3202K	1	I12
Terminal 18	Marrón-Verde	DDI 3202K	1	I13
Terminal 19	Blanco-Amarillo	DDI 3202K	1	I14
Terminal 20	Amarillo-Marrón	DDI 3202K	1	I15
Terminal 21	Blanco	DDI 3202K	2	I16
Terminal 22	Marrón	DDI 3202K	2	I17

Tabla 4.5 Conexionado de cables de las entradas del 1º DB con el autómat

De esta forma, las entradas del 1º DB se encuentra conectadas en su totalidad al autómat.

Por su parte, el cableado del 2º DB es diferente ya que contiene tanto entradas como salidas. Para las entradas, el cableado sí será igual que las del 1º DB, es decir, realizando cableado directo al automático, quedando de la siguiente manera:

TERMINAL	COLOR	MÓDULO	MANGUERA	CANAL
Terminal 5	Verde	DDI 3202K	2	I18
Terminal 6	Amarillo	DDI 3202K	2	I19
Terminal 7	Gris	DDI 3202K	2	I20
Terminal 8	Rosa	DDI 3202K	2	I21
Terminal 9	Azul	DDI 3202K	2	I22
Terminal 10	Rojo	DDI 3202K	2	I23
Terminal 11	Negro	DDI 3202K	2	I24
Terminal 12	Morado	DDI 3202K	2	I25

Tabla 4.6 Conexión de cables de las entradas del 2º DB con el automático

Con este cableado, quedan conectadas todas y cada una de las entradas, por lo que los colores restantes de la manguera 2 del BMX DDI 3202K que se utilicen para entradas y los colores de que pertenezcan al BMX DDM 16022 que también sirvan para ese fin no tendrán conexión de ningún tipo.

Así pues, ya utilizados todos los cables del módulo BMX DDI 3202K en lo referido a entradas, se procede a la conexión de los cables que sirven para la alimentación del propio módulo. Estos son los cables Blanco-Gris, Marrón-Gris, Blanco-Rosa y Gris-Rosa de ambas mangueras del módulo, haciendo un total de 8 cables.

La conexión de la alimentación de estas 2 mangueras se realiza de la siguiente manera:

- Se realiza un puente entre los cables Gris-Blanco y Rosa-Blanco de la manguera 1, otro puente entre los mismos colores pero de la manguera 2 y, por último, otro puente entre los 2 cables Gris-Marrón y los 2 Rosa-Marrón, es decir, un total de 4 cables.

- Una vez puenteados, los 2 cables Gris-Marrón y los 2 Rosa-Marrón se sueldan con un cable de color negro y los cables Gris-Blanco y Rosa-Blanco lo harán con un cable de color gris. Esto se realiza para trabajar con un único cable que además tenga una dimensión mayor para un mejor manejo, quedando finalmente 1 cable negro y dos cables grises.

- El cable negro, se conecta directamente al negativo de la fuente de alimentación de 24 VCC. Por su parte, cada uno de los cables grises, irá conectado a un fusible de 0.5 mA, y posteriormente, al positivo de la fuente de 24 VCC.

Después de haber realizado la conexión de todas las entradas, es el turno de la conexión de las salidas. En primer lugar, se conectarán las salidas de los motores, los cuales se sitúan en los terminales del 2º DB y están conectados al común de los relés. Para su conexión, se utilizarán cables de color azul. Así pues, la conexión quedaría de la siguiente manera:

TERMINAL DB	TERMINAL RELÉ (C)
Terminal 2	4I
Terminal 3	4J
Terminal 15	4K
Terminal 16	4L

Tabla 4.7 Conexión de los motores con sus terminales correspondientes

Por su parte, tal y como se pudo observar en la Figura 4.6, todos los NA están conectados al menos (-) de la fuente de alimentación de 12 VCC, por lo que se realiza un puente entre todos ellos y se conecta finalmente al negativo de la fuente. El color de cable escogido para realizar esta conexión es el color negro. Lo mismo ocurrirá con todos los NC, cuyos cables se encuentran puenteados entre sí finalizando en el positivo de la fuente de 12 VCC. Estos cables serán de color gris.

Una vez realizada la conexión de los motores, se pasa a la conexión de los leds. Ya que sólo se dispone de 4 salidas para ellos en el módulo BMX DDI 16022, se ha optado por la conexión de los leds de la planta 0 y 3, y los leds de subida de las plantas 1 y 2 para que, al menos, pueda visualizarse un led por planta.

Igualmente, como se puede apreciar en la Figura 4.7, el común de todos los leds están conectados al negativo de la fuente de 24 VDC, por lo que se opta por puentear todos los comunes y estos, a su vez, conectarlos al negativo de la fuente, quedando la conexión (1M→1N→1O→1P→-24VCC). Por su parte, los NA de estos relés están conectados a los cátodos de los leds, quedando la siguiente conexión:

TERMINAL DB	TERMINAL RELÉ (NA)
Terminal 17	2M
Terminal 18	2N
Terminal 19	2O
Terminal 22	2P

Tabla 4.8 Conexión de los leds con sus terminales correspondientes

No se debe olvidar, que entre un terminal y otro es preciso la instalación de una resistencia de 2 K Ω para evitar que se quemen.

Ya conectados los motores y los leds, tan solo quedan dos terminales del 2º DB sin conectar. Estos terminales son los terminales 1 y 14 que corresponden al común de las entradas y al común de los leds (ánodos), respectivamente. Estos terminales están conectados al positivo (+) de la fuente de alimentación, para así permitir cerrar sus respectivos circuitos.

Tras esto, la conexión de todos los cables de la maqueta se encuentra realizada quedando, solamente, la conexión de las salidas y de la alimentación del módulo BMX DDM 16022. Las salidas irán conectadas a las bobinas de los relés, ya que en el momento en que se cambie de estado binario, será la bobina la encargada de realizar la conmutación del relé. Así pues, las salidas de este módulo irán conectadas a los terminales de las bobinas de los relés de la siguiente forma:

TERMINAL	COLOR	MÓDULO	CANAL
Terminal I	Negro	DDM 16022	Q16
Terminal J	Morado	DDM 16022	Q17
Terminal K	Gris-Rosa	DDM 16022	Q18
Terminal L	Azul-Rojo	DDM 16022	Q19
Terminal M	Blanco-Verde	DDM 16022	Q20
Terminal N	Marrón-Verde	DDM 16022	Q21
Terminal O	Blanco-Amarillo	DDM 16022	Q22
Terminal P	Marrón-Amarillo	DDM 16022	Q23

Tabla 4.9 Conexión de las salidas

El otro lado de cada una de las bobinas, están concentrados en el Terminal (—), el cuál debe estar conectado al negativo de la fuente de alimentación de 24 VCC para, igualmente, cerrar el circuito de las bobinas.

Finalmente, lo único que queda por instalar son aquellos cables que se encargan de la alimentación del módulo BMX DDM 16022. Para ello, los cables Marrón-Gris y Marrón-Rosa, pertenecientes a los sensores 9 y 19 de este módulo respectivamente, se conectarán al negativo de la fuente de alimentación de 24 VCC. Por su parte, el cable Blanco-Gris (sensor 10 del módulo), debe conexionarse al fusible de 0.5 mA y posteriormente, al positivo de la fuente de 24 VCC, ya que es la alimentación de las entradas del módulo. Por último, el cable Blanco-Rosa (sensor 20) se conexionará en primer lugar al fusible de 2 mA, al ser la alimentación de las salidas, para después hacerlo al positivo de la fuente de alimentación.

Con todo esto, se completa el circuito de conexionado maqueta-autómata quedando de la siguiente manera.

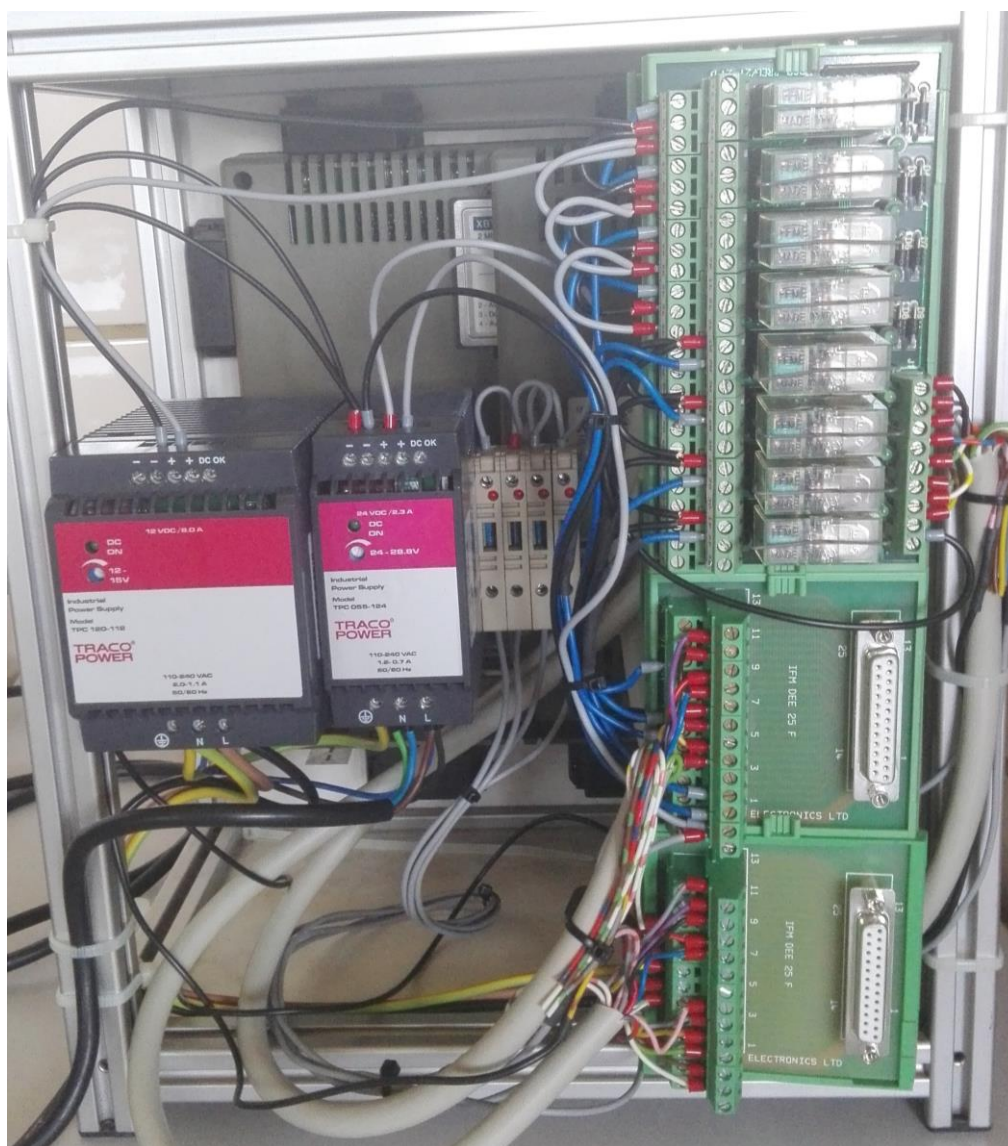


Figura 4.8 Circuito de conexionado del autómatas

Como resumen y para visualizarlo de una forma más amena se adjunta a continuación, un esquema del conexionado final, excluyendo la conexión a la red eléctrica

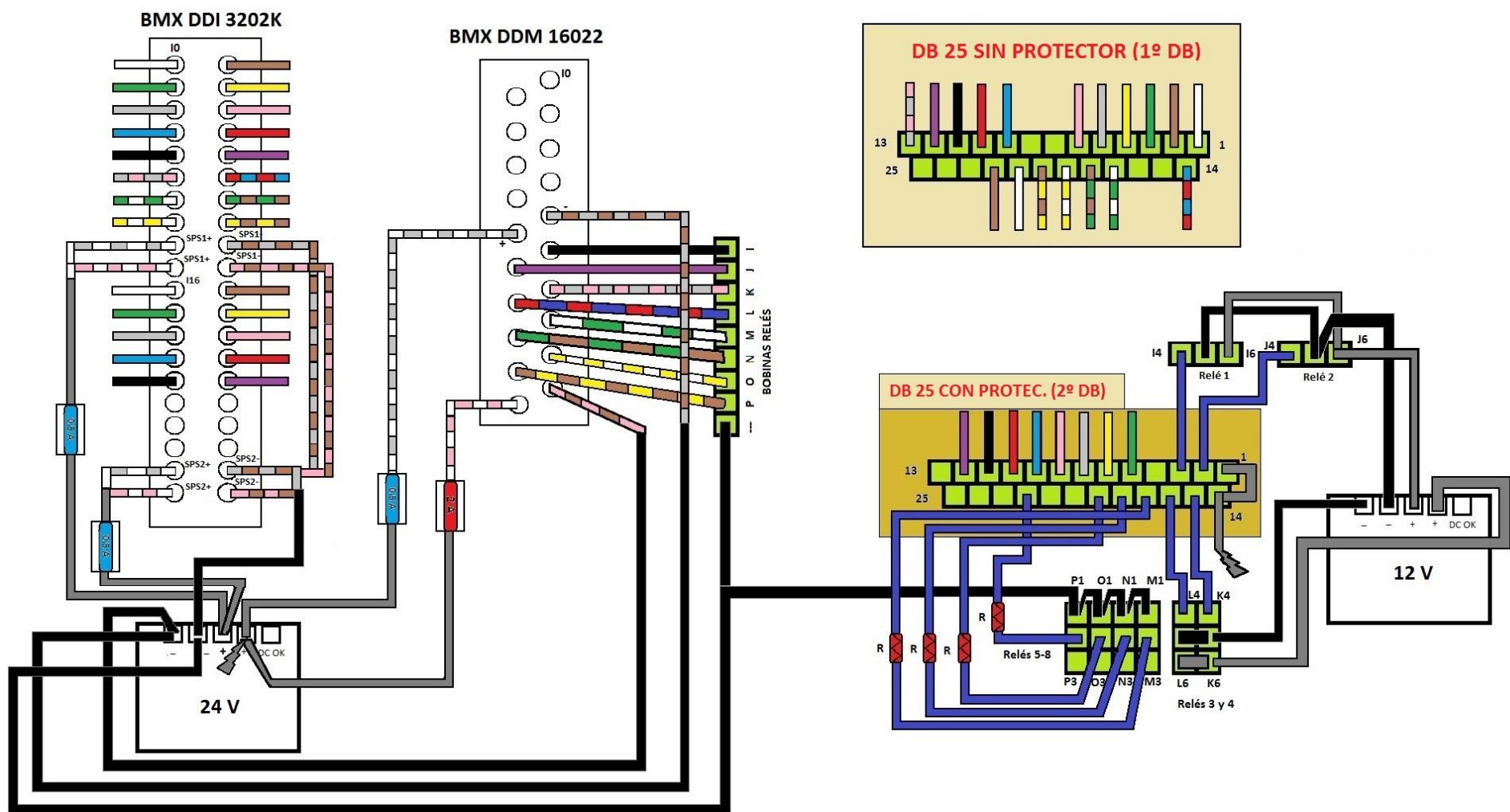


Figura 4.9 Esquema general de conexionado maqueta-autómata

5 CoDeSYS

CoDeSys es un software informático creado por 3S-Smart Software Solutions que nos permite programar autómatas de diferentes fabricantes, así como generar código para algunos microprocesadores y DSPs. Sigue el estándar industrial IEC 61131-3, el cual permite la utilización de diferentes lenguajes de programación y bloques funcionales a través de los llamados POU. Además de la programación, CoDeSys permite la visualización de dicha programación, siempre y cuando se realice una recreación del sistema a automatizar, con el fin de poder observar el resultado final del sistema [24].

Codesys permite programar en los diferentes lenguajes que se encuentran en el estándar IEC 61131-3:

- **SFC**: se trata de un método gráfico de modelado y descripción de sistemas de automatismos secuenciales, en los que el estado que adquiere el sistema ante el cambio de una entrada depende de los estados anteriores. Se deriva de sus antecesores “Redes de Petri”. Se trata de programas que están bien estructurados y cuyos elementos básicos son las etapas, las acciones y las transiciones. De este modo, una secuencia en SFC se compone de una serie de etapas representadas por cajas rectangulares y que se encuentran conectadas entre sí por líneas verticales. Así, cada etapa representa un estado particular del sistema y cada línea vertical a una transición. Estas transiciones están asociadas a una condición ‘verdadero/falso’, dando paso así a la desactivación de la etapa que la precede y activación de la posterior [25].

- **FBD**: es un lenguaje de tipo gráfico que permite al usuario programar expresiones y lógica booleana. Su utilización es conveniente cuando no hay ciclos pero existen, sin embargo, varias ramas en el programa a crear. Se trata de un lenguaje de alto nivel que permite resumir funciones básicas en bloques de modo que el usuario solo se preocupa por una programación funcional de su rutina.

- **LD**: Se trata de una conexión gráfica entre variables de tipo Booleano donde se representa el flujo de energía en diagramas de circuitos eléctricos. Dentro de sus características principales se encuentra el uso de barras de alimentación y elementos de enlace y la posibilidad de utilizar contactos y bobinas. Su estructura es simple, ya que en el momento en que se cumple la condición que se espera en el contacto de la izquierda (que equivale al cumplimiento de la condición de entrada), se produce el paso del flujo de corriente que lo conecta a la bobina de la salida, permitiendo la realización de dicha acción. En el momento en que deja de cumplirse la acción de la entrada, el flujo de corriente deja de circular.

- **ST**: está basado en los lenguajes texto de alto nivel y es similar a C pero, sobre todo, PASCAL, por lo que requiere conocimiento previo de programación. Las principales ventajas de este lenguaje es que incluye la formulación de las tareas del programa, una clara construcción de los programas en bloques con reglas (instrucciones) y una potente construcción para el control. De este modo, se trata de la forma más apropiada de programar cuando queremos realizar ciclos [26].

- **IL**: es un lenguaje de bajo nivel, similar al lenguaje ensamblador. Con IL solo una operación es permitida por línea. Este lenguaje es adecuado para pequeñas aplicaciones y para optimizar partes de una aplicación.

Además de estos lenguajes de programación, CoDeSys incluye un lenguaje adicional, el **CFC** el cual está orientado a FBD y permite interconectar gráficamente las funciones complejas y existentes, por lo que no es necesario programar numerosas funciones estándar, ya que se dispone de librerías [27].

5.1 Introducción al Software

El software informático CoDeSys presenta una pantalla de explotación en la que encontramos los siguientes elementos:

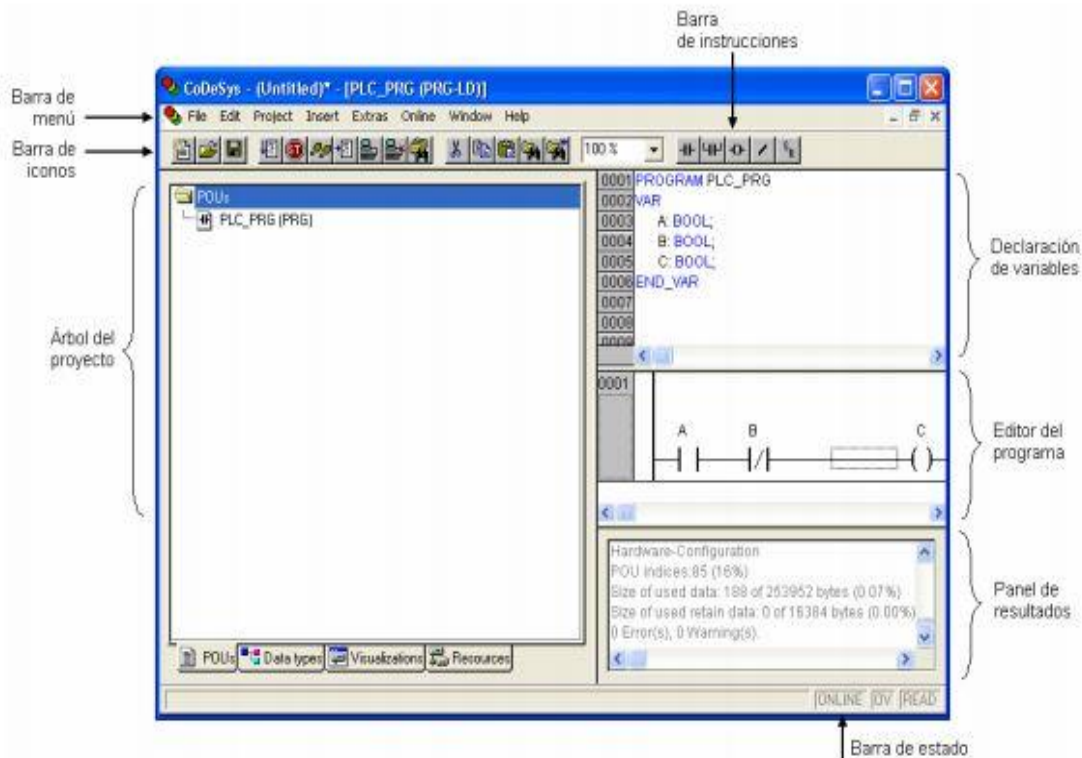


Figura 5.1 Ventana principal del programa CoDeSys

- **Barra de menú:** Permite realizar las acciones fundamentales de cualquier software informático
- **Barra de iconos:** Contiene accesos directos a las acciones fundamentales más comunes dentro del programa.
- **Barra de instrucciones:** Contiene accesos a las herramientas más utilizadas para realizar acciones dentro de cada tipo de lenguaje de programación. En la imagen, se aprecian las herramientas del lenguaje LD.
- **Declaración de variables:** Es una ventana en la cual se realiza la declaración de las variables locales de la POU en cuestión.
- **Editor del programa:** Contiene el texto o simbología propia utilizada para realizar las acciones que contiene cada POU. Para añadir interacciones en dicho editor se utilizan las herramientas de la barra de instrucciones.
- **Panel de resultados:** Esta pestaña muestra la información básica de la memoria utilizada, así como los errores producidos a la hora de la programación. Para que esta información sea apreciable se debe hacer clic en **Project** y a continuación en la opción **Build**, si se quiere comprobar el estado de la POU actual, o **Rebuild All**, para comprobar todo el proyecto.
- **Barra de estado:** Nos informa del estado actual del programa, es decir, si nos encontramos ONLINE/OFFLINE respecto al autómat, si el programa se encuentra en ejecución, etc.
- **Árbol del proyecto:** Es una ventana que nos permite acceder a las cuatro funciones principales del programa. La pestaña POU's se utiliza para la visualización y creación de las nuevas POU's; Data types permite definir al usuario sus propios tipos de datos; Visualization permite recrear mediante imágenes y formas la programación realizada; y Resources contiene toda la información tanto de las variables globales como la del PLC usado [28].

El primer paso para comenzar a programar es la creación de un nuevo proyecto. Para ello, se debe hacer clic en **File** y seguidamente en la acción **New**. Una vez realizado esto aparecerá una pantalla como el de la Figura 5.2, donde se debe elegir el autómata al cual nos vayamos a conectar. En el caso de este proyecto, ya que el programa CoDeSys se encargará de realizar únicamente simulaciones se elegirá la opción **None**.



Figura 5.2 Menú selección de autómata

Una vez elegido esto, aparecerá una nueva pantalla en la que se debe elegir el primer POU. Por defecto, aparecerá con el nombre PLC_PRG con POU de tipo Program, el cual no se variará ya que será el encargado de ejecutar todas las POUs que se creen dentro del programa a ejecutar. El tipo de lenguaje recomendado para esto es el lenguaje ST.

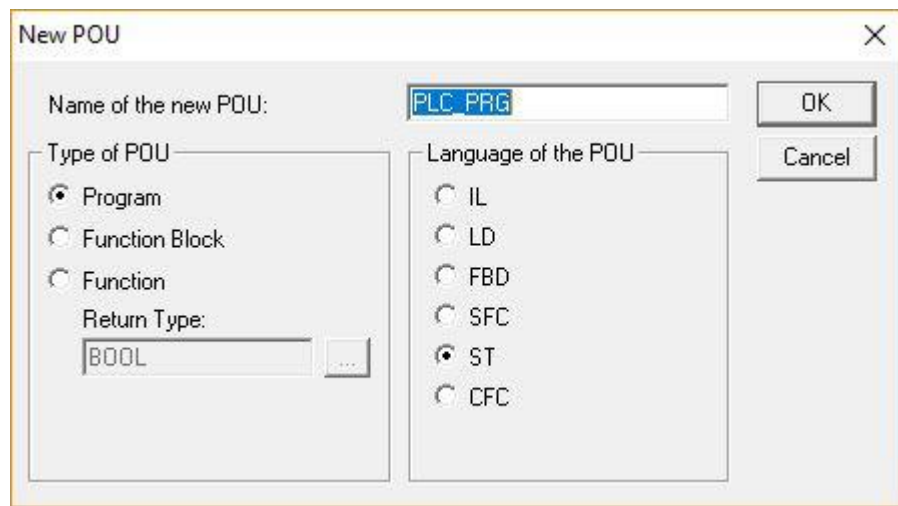


Figura 5.3 Menú de configuración de la POU principal

Una vez creada la POU de ejecución, se pasa a la creación de nuevas POUs, que tendrán la función de realizar las distintas maniobras de la simulación. Para añadir una nueva POU basta con hacer clic en el botón derecho sobre el icono de la carpeta con nombre **POUs** y, a continuación, hacer clic en el botón **Add Object**. Se podrán crear de esta manera todas las POUs que se deseen. Como se hizo con la primera, se seleccionará el tipo de POU, el lenguaje utilizado y el nombre escogido para dicha POU.

Para crear una visualización que permita observar aquello que se ha programado mediante botones, imágenes en movimiento, etc., debemos acceder a la pestaña Visualizations, en la carpeta Visualization hacer clic en el botón derecho y, seguidamente, **Add Object**. Con esto, solo quedará seleccionar el nombre deseado.

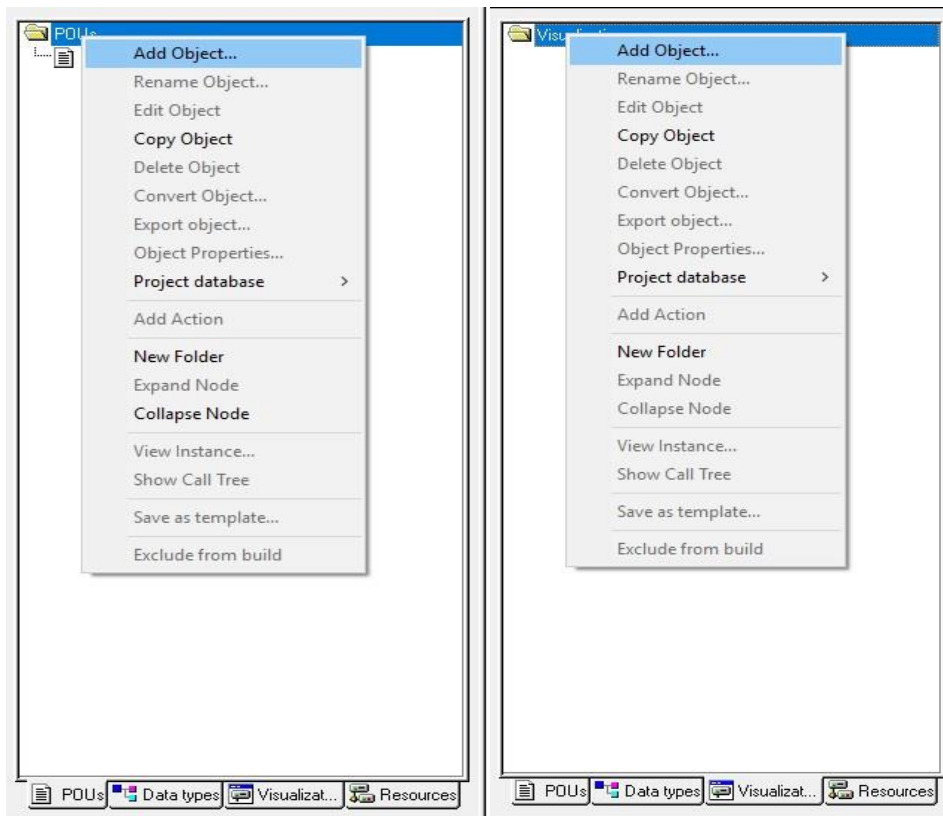


Figura 5.4 A la izquierda, ventana de creación de una nueva POU. A la derecha, ventana de creación de una visualización

Por último, todas las variables utilizadas en los programas han de ser declaradas. Si las variables utilizadas son variables locales pueden ser declaradas en la misma POU, mientras que si las variables son globales para diferentes POU's deben ser declaradas como variables globales. Para ello, se debe acceder a **Resources** → **Global Variables** → **Global_Variables**. CoDeSys nos permite la creación de variables de distintos tipos: BOOL, INT, LREAL, ARRAY, BYTE, DATA, STRING, etc.

5.2 Simulación de dos ascensores de 7 plantas con maniobra universal

En la simulación de esta primera maniobra, la maniobra universal para 2 ascensores, se ha optado por utilizar tres tipos de lenguajes diferentes: SFC, LD y ST.

El Lenguaje SFC es utilizado para el movimiento general del ascensor, es decir, la secuencia que seguirá cada ascensor desde que se produce una llamada por parte de un usuario hasta que el ascensor se queda libre o bien recibe otra llamada tras haber realizado la anterior.

El lenguaje LD ha sido utilizado para la activación de los temporizadores que mantendrán las puertas abiertas y para activar otras variables que permitirán continuar con el transcurso de la acción completa del ascensor.

Por último, el lenguaje ST ha permitido la modificación de las variables que permiten que la recreación realizada pueda seguirse en tiempo real mediante una visualización. Igualmente, con este lenguaje se han realizado todas las casuísticas en función de la posición del ascensor y cuál debe atender una llamada, si las llamadas son realizadas desde el interior o desde el exterior, impedir que un ascensor atienda otras llamadas, etc.

Todo esto será comentado más en profundidad en los siguientes subcapítulos.

5.2.1 Lenguaje SFC

Como bien se ha comentado, este lenguaje se encarga de realizar la secuencia completa del movimiento del ascensor, con su respectiva apertura y cierre de puertas.

La secuencia comienza con ambos ascensores en la planta baja. Nada más comenzar la simulación se sitúa el segundo ascensor en una planta intermedia (5º planta) para evitar que, si se realiza una llamada desde uno de los pisos superiores, el usuario esté demasiado tiempo esperando.

Cuando el ascensor recibe una llamada, automáticamente dicho ascensor pasa a un estado de ocupado y no podrá atender otra llamada hasta que haya completado la actual. Si el ascensor se encuentra en una planta superior a la de la llamada, el ascensor bajará; mientras que si la planta actual es inferior a la de llamada, el ascensor subirá. Una vez llegue a su planta, o la llamada haya sido realizada mientras el ascensor se encuentra en ese mismo piso, el ascensor abrirá sus puertas. En ese momento, el ascensor queda liberado para una posible llamada interna, dando así prioridad de uso a alguien que ya se encuentre dentro de la cabina.

Cuando el ascensor haya abierto las puertas en su totalidad se activará un temporizador de 5 segundos que simulan la entrada o salida de las personas de la cabina del ascensor. Si se cumple este tiempo, o algún usuario antes de cumplirse presiona el botón de cerrar desde dentro de la cabina, las puertas comenzarán a cerrarse.

En el momento en que se cierran se activa un nuevo temporizador de 3 segundos, que permitirá al usuario que haya entrado en la cabina o que se encuentre dentro de ella poder escoger su piso de destino. Aunque se permita la pulsación interna desde el momento en que se abren las puertas, lo natural es que el usuario que entre en el ascensor seleccione su piso de destino con más demora, llegando incluso a cerrarse las puertas sin haber realizado su selección, de ahí este tiempo extra de 3 segundos.

Si en estos 3 segundos el ascensor no recibe ninguna llamada desde la botonera de la cabina, el ascensor quedará liberado completamente para ser utilizado tanto desde el interior como desde el exterior y comenzará de nuevo el ciclo a la espera de una nueva llamada. Si por contra se realiza una llamada desde el interior, el ascensor reconocerá esta petición y comenzará su desplazamiento hacia el piso de destino de la misma forma que antes.

Si pasados 40 segundos no se produce ninguna llamada en general, el ascensor 1 adquiere una llamada para situarse en la planta baja y el ascensor 2 adquiere otra para situarse en la 5º planta, volviendo a permitir que en caso de una llamada posterior, los ascensores tarden el menor tiempo posible en llegar a cualquier planta.

En caso de producirse una parada de algún ascensor (ya sea simulando un corte de luz o un error), el ascensor detendrá sus movimientos. Si cuando se restaure el movimiento del ascensor, su llamada ha sido trasladada al otro ascensor, el ascensor que estuvo parado volverá a su planta inicial (1º planta en caso del ascensor 1, 5º planta en caso del ascensor 2). Sin embargo, si el ascensor mantiene su llamada, éste continuará su curso tal y como estaba realizando. La casuística para que un ascensor traslade al otro su llamada se reflejan en el lenguaje ST.

Así pues, un grafect general de esta secuencia de acciones del ascensor 1 es el siguiente (para el ascensor 2, simplemente cambia el traslado a la planta 5 en lugar de la planta 1).

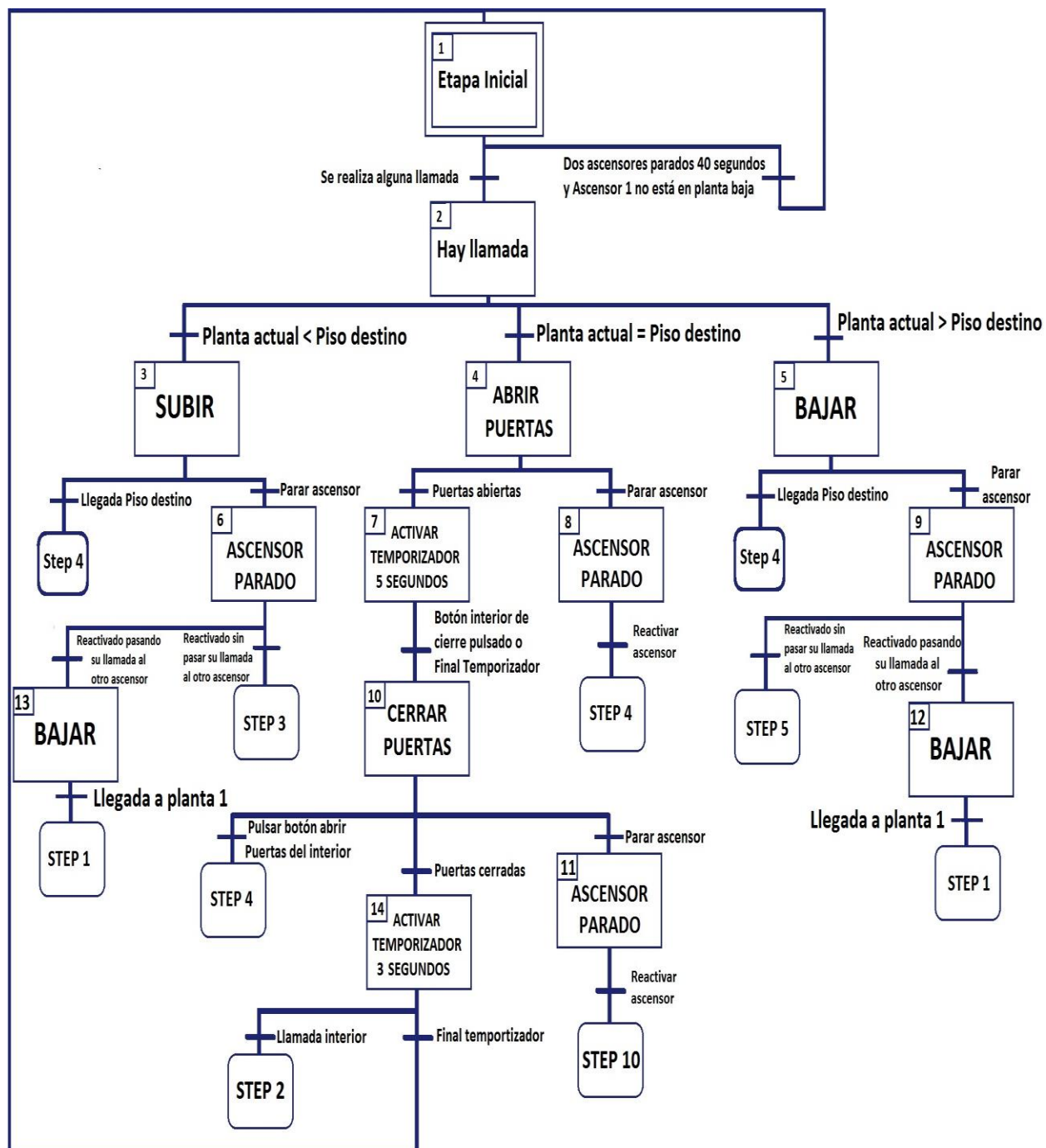


Figura 5.5 Graficet general del Ascensor 1 con maniobra universal

5.2.2 Lenguaje LD

Este lenguaje contiene los temporizadores tanto de 5 como de 3 segundos de ambos ascensores, que se pondrán en funcionamiento cuando se activen sus respectivos sensores y que, al finalizar, activarán el actuador que tenga cada uno asociado. Se pueden encontrar también utilizados en este lenguaje dos temporizadores más:

- Un temporizador de 40 segundos que se activará cuando los dos ascensores se encuentren sin llamadas. Al finalizar el temporizador se dará la orden de recolocar a los ascensores en sus plantas por defecto.
- Un temporizador de 10 segundos, el cual se activa cuando uno de los dos ascensores se encuentra en parada forzada y que, al cumplirse este tiempo, activa una variable encargada de traspasar la llamada del ascensor parado al otro ascensor en el caso de tener realizada alguna petición y ser posible su traspaso.

Además de esto, se incluye una acción que impide la selección de un piso interior hasta que las puertas de las cabinas no se abran por primera vez. Esto se realiza para que sea imposible realizar una llamada desde el interior en el momento en que se arranca la simulación, ya que es imposible que haya alguna persona dentro de la cabina.

También, se encuentra una acción que permite cerrar las puertas según haya terminado el temporizador de 5 segundos o se haya accionado el pulsador de cerrar puertas desde el interior.

5.2.3 Lenguaje ST

Se podría decir que en este lenguaje se encuentra recogidas las entrañas de la maniobra ya que, al estar declarados todos los pulsadores y sensores mediante vectores (ARRAY), es necesario estar realizando ciclos constantes para conocer el estado de todos estos pulsadores y sensores de plantas. Así pues, en este lenguaje se halla la programación necesaria para las siguientes casuísticas:

- En caso de que ninguno de los ascensores se encuentren en parada forzada, si ambos ascensores están libres, aquel que debe atender la llamada será el que se encuentre más cerca del piso de destino. Si por contra, hay tan solo un ascensor libre y disponible, la llamada será atendida por dicho ascensor.
- Si ningún ascensor se encuentra disponible para atender una llamada, se encenderán todas las luces de los pasillos para hacerlo saber a los usuarios
- En el momento en que un ascensor llega a su destino y comiencen a abrirse sus puertas, la luz del interior que se encuentre encendida se apagará, permitiendo así realizar una nueva llamada desde la cabina.
- Todas las luces de los pasillos se apagarán cuando ambos ascensores estén libres, si uno de ellos está libre y el otro ocupado pero con una llamada interior, o si alguno de los ascensores está vacío y el otro mantiene una parada forzosa. Esto permitirá realizar una llamada desde la botonera de pasillo de cualquier planta.
- Cuando en un ascensor se produzca una llamada desde el interior, siempre que el ascensor no esté en ejecución, se encenderá la luz del interior y se impedirá que se realice otra llamada diferente, ya que en ese momento el ascensor tendrá asignado un destino.
- La planta actual de cada ascensor será actualizada por cada paso del sensor de planta.
- Si un ascensor se encuentra en parada forzada y estaba en ejecución de una llamada de pasillo, si se han cumplido los 10 segundos correspondientes a encontrarse parado y el otro ascensor está libre, éste ascensor libre adquirirá la llamada del ascensor parado con el fin de atender la llamada realizada y pendiente.
- En el momento en que un ascensor entre en estado de parada forzada, el otro ascensor atenderá todas las llamadas.
- Por último, se realizará la comparación en todo momento de la localización de la cabina con la del piso de destino con el objetivo de conocer si el movimiento que el ascensor debe realizar para atender la llamada es de subida, bajada o mantenerse parado al encontrarse en el piso de destino.

Además de estos casos, gracias a otra POU en lenguaje ST se realizan los movimientos de subida y bajada de cada ascensor, abrir y cerrar las puertas, reconocer la posición del ascensor en función de su altura u observar a modo de display el piso en el cual se encuentra el ascensor en cada momento, entre otros. Esta última POU será la encargada de permitirnos seguir visualmente los distintos movimientos, para la cual disponemos de la siguiente visualización.

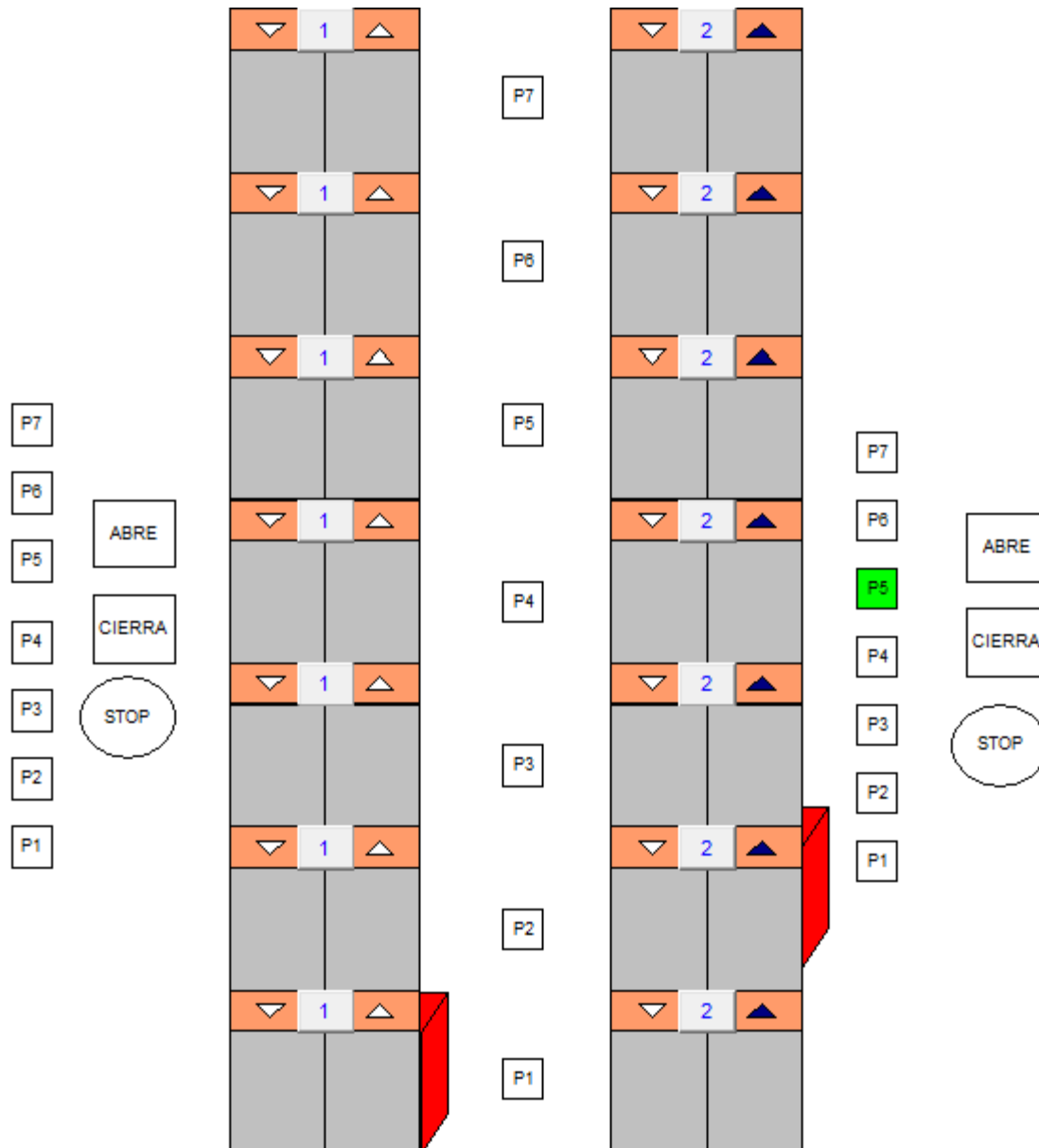


Figura 5.6 Imagen de la visualización de los ascensores con maniobra universal

Por último, las variables globales utilizadas para entender mejor el funcionamiento del programa son las siguientes:

```

0001 VAR_GLOBAL
0002
0003 (* ASCENSOR 1 y 2 *)
0004
0005 AMBOS_VACIOS: BOOL; (*Variable que se activa cuando ambos ascensores están vacíos y han pasado 40 segundos*)
0006 ALGO_PULSADO: INT; (*Adquiere el nº del piso llamado y permite conocer la asignación del ascensor*)
0007 FT10: BOOL; (*Indica la finalización del temporizador de 10 segundos*)
0008 PULSADOR_EXT: ARRAY [1..10] OF BOOL; (*Pulsadores de cada planta exterior de ambos ascensores*)
0009 LUZ: ARRAY[1..10] OF BOOL; (*Luz de los pulsadores exteriores *)
0010 START: BOOL:=TRUE; (*Variable que permite que, al comenzar, ambos ascensores ocupen sus plantas de reposo*)
0011
0012
0013 (*ASCENSOR 1*)
0014
0015 AT3: BOOL; (*Permite el comienzo del temporizador de 3 segundos*)
0016 FIN_TEMP3: BOOL; (*Indica la finalización del temporizador de 5 segundos*)
0017 AT5: BOOL; (*Permite el comienzo del temporizador de 5 segundos*)
0018 FIN_TEMP5: BOOL; (*Indica la finalización del temporizador de 5 segundos*)
0019 FLAG: BOOL; (*Variable auxiliar que permite accionar un pulsador interior cuando se abran las puertas por 1ª vez*)
0020
0021 IGUAL: BOOL; (*Indica que la posición de la cabina es igual al destino*)
0022 MAYOR: BOOL; (*Indica que la posición de la cabina es mayor al destino*)
0023 MENOR: BOOL; (*Indica que la posición de la cabina es menor al destino*)
0024
0025 BAJAR: BOOL; (*Al activarse hace que el ascensor 1 comience a bajar*)
0026 SUBIR: BOOL; (*Al activarse hace que el ascensor 1 comience a subir*)
0027 STOP1: BOOL; (*Estado de parada forzada*)
0028
0029 BLOQ_INTERNA: BOOL; (*Impide una llamada interior*)
0030 LLAMADA_INTERNA: BOOL; (*Se activa cuando se produce una llamada interior en el ascensor 1*)
0031 PULSADOR_INT1: ARRAY [1..10] OF BOOL; (*Pulsadores interiores del ascensor*)
0032 LUZ_INT: ARRAY[1..10] OF BOOL; (*Luz de los pulsadores interiores*)
0033 PCP: BOOL; (*Pulsador cerrar puertas*)
0034 PAP: BOOL; (*Pulsador abrir puertas*)
0035
0036 EMPIEZA: BOOL; (*Indica el retorno del ascensor cuando se produce un traspaso de llamadas al otro ascensor*)
0037 OCUPADO: BOOL; (*Indica que el ascensor no está disponible para ser llamado por un usuario de pasillo*)
0038
0039 PUERTA_DER: ARRAY [1..10] OF LREAL; (*Movimiento de la puerta derecha de cada planta*)
0040 PUERTA_IZQ: ARRAY [1..10] OF LREAL; (*Movimiento de la puerta izquierda de cada planta*)
0041 MAX_APE: ARRAY [1..10] OF BOOL; (*Maxima apertura de cada puerta*)
0042 MAX_CIE: ARRAY [1..10] OF BOOL; (*Maximo cierre de cada puerta*)
0043 MAXIMA_APERTURA: BOOL; (*Las puertas del piso en el que se encuentra la cabina están abiertas completamente*)
0044 MAXIMO_CIERRE: BOOL; (*Indica que todas las puertas se encuentran cerradas completamente*)
0045 COND_CERRAR: BOOL; (*Indica que se ha producido una de las condiciones que permite cerrar las puertas*)
0046 ABRIENDO_PUERTAS: BOOL; (*Estado del ascensor mientras abre las puertas*)
0047 CERRANDO_PUERTAS: BOOL; (*Estado del ascensor mientras cierra las puertas*)
0048
0049 PLANTA: INT; (*Indica la planta actual*)
0050 PISO_PULSADO: INT; (*Indica el piso de destino*)
0051 SENSOR: ARRAY [1..10] OF BOOL; (*Sensores de cada planta del ascensor*)
0052 DISPLAY1: INT; (*Muestra en el gráfico el piso por el que transcorre la cabina*)
0053
0054
0055 (*ASCENSOR 2*)
0056
0057 AT32: BOOL; (*Permite el comienzo del temporizador de 3 segundos*)
0058 FIN_TEMP32: BOOL; (*Indica la finalización del temporizador de 3 segundos*)
0059 AT52: BOOL; (*Permite el comienzo del temporizador de 5 segundos*)
0060 FIN_TEMP52: BOOL; (*Indica la finalización del temporizador de 5 segundos*)
0061 FLAG2: BOOL; (*Variable auxiliar que permite accionar un pulsador interior cuando se abran las puertas por 1ª vez*)
0062
0063 IGUAL2: BOOL; (*Indica que la posición de la cabina es igual al destino*)
0064 MAYOR2: BOOL; (*Indica que la posición de la cabina es mayor al destino*)
0065 MENOR2: BOOL; (*Indica que la posición de la cabina es menor al destino*)
0066
0067 BAJAR2: BOOL; (*Al activarse hace que el ascensor 2 comience a bajar*)
0068 SUBIR2: BOOL; (*Al activarse hace que el ascensor 2 comience a subir*)
0069 STOP2: BOOL; (*Estado de parada forzada*)
0070
0071 BLOQ_INTERNA2: BOOL; (*Impide una llamada interior*)
0072 LLAMADA_INTERNA2: BOOL; (*Se activa cuando se produce una llamada interior en el ascensor 2*)
0073 PULSADOR_INT2: ARRAY [1..10] OF BOOL; (*Pulsadores interiores del ascensor*)
0074 LUZ_INT2: ARRAY[1..10] OF BOOL; (*Luz de los pulsadores interiores*)
0075 PAP2: BOOL; (*Pulsador abrir puertas*)
0076 PCP2: BOOL; (*Pulsador cerrar puertas*)
0077
0078 EMPIEZA2: BOOL; (*Indica el retorno del ascensor cuando se produce un traspaso de llamadas al otro ascensor*)
0079 OCUPADO2: BOOL; (*Indica que el ascensor no está disponible para ser llamado por un usuario de pasillo*)
0080
0081 PUERTA_DER2: ARRAY [1..10] OF LREAL; (*Apertura y cierre de la puerta derecha de cada planta*)
0082 PUERTA_IZQ2: ARRAY [1..10] OF LREAL; (*Apertura y cierre de la puerta izquierda de cada planta*)
0083 MAX_APE2: ARRAY [1..10] OF BOOL; (*Maxima apertura de cada puerta*)
0084 MAX_CIE2: ARRAY [1..10] OF BOOL; (*Maximo cierre de cada puerta*)
0085 MAXIMA_APERTURA2: BOOL; (*Las puertas del piso en el que se encuentra la cabina están abiertas completamente*)
0086 MAXIMO_CIERRE2: BOOL; (*Indica que todas las puertas se encuentran cerradas completamente*)
0087 COND_CERRAR2: BOOL; (*Indica que se ha producido una de las condiciones que permite cerrar las puertas*)
0088 ABRIENDO_PUERTAS2: BOOL; (*Estado del ascensor mientras abre las puertas*)
0089 CERRANDO_PUERTAS2: BOOL; (*Estado del ascensor mientras cierra las puertas*)
0090
0091 PLANTA2: INT; (*Indica la planta actual*)
0092 PISO_PULSADO2: INT; (*Indica el piso de destino*)
0093 SENSOR2: ARRAY [1..10] OF BOOL; (*Sensores de cada planta del ascensor*)
0094 DISPLAY2: INT; (*Permite conocer el piso en el que la cabina se encuentra*)
0095
0096 END_VAR

```

Figura 5.7 Variables globales utilizadas en la simulación de los ascensores con maniobra universal

5.3 Simulación de dos ascensores de 7 plantas con maniobra colectiva selectiva

Al igual que ocurriese con la maniobra universal, para el funcionamiento de este tipo de ascensores se utilizarán los mismos lenguajes de programación y cada uno de ellos también servirán para realizar las mismas funciones, es decir, el lenguaje SFC nos permitirá seguir las acciones que ocurrirán en una secuencia completa del movimiento del ascensor; el lenguaje LD será utilizado para la activación de los temporizadores y variables necesarias para el correcto funcionamiento del ascensor; y el lenguaje ST se utiliza para los diferentes casos y situaciones que pueden darse cuando se produce una llamada del ascensor, además de incluir los movimientos de subida y bajada del ascensor, la apertura y cierre de puertas, etc., que permitirán seguir la simulación de una forma más sencilla.

Cabe destacar que para la simulación de esta maniobra se ha realizado una pequeña variación con la real. En el momento en que el ascensor se dirige hacia un piso de destino cuyo propósito del usuario es ir en la dirección contraria a la que debe realizar el ascensor para atenderla, las llamadas realizadas en el trayecto hasta la recogida del usuario que sean de plantas que se encuentren entre la planta actual de la cabina y la del piso de destino no serán atendidas aun siendo llamadas para realizar un desplazamiento similar al que está realizando el ascensor. Esto es si, por ejemplo, un usuario realiza una llamada desde el pasillo de la planta 2 para ascender y la cabina se encuentra en una planta superior a esta, el ascensor descenderá. Si en ese trayecto hasta llegar a la planta 2 el ascensor recibe alguna llamada de un piso intermedio para bajar hacia alguna planta, dicha llamada se obviará hasta que sea posible su realización. Aunque teóricamente debería atender dicha llamada ya que se produce en el mismo sentido del movimiento del ascensor (bajar), en este caso se ha decidido que será obviada para primar aquellas peticiones que llevan mayor tiempo de espera.

5.3.1 Lenguaje SFC

Al igual que ocurriera con el sistema de ascensores con maniobra universal, la secuencia comienza con los dos ascensores en la planta baja, instalándose nada más comenzar la simulación el ascensor 2 en la 5ª planta.

En el instante en que el ascensor recibe la primera llamada, éste compara su posición actual con la localización del piso de destino y en función de si es un piso superior, inferior o el mismo, el ascensor subirá, bajará o se mantendrá en el mismo punto, respectivamente.

Al utilizarse una maniobra colectiva selectiva en subida y bajada, el ascensor podrá memorizar más de una llamada, a diferencia del sistema con maniobra universal y, por tanto, si se ha realizado una llamada de un piso intermedio que tiene como dirección la misma que el piso de destino, al llegar a dicha planta intermedia, el ascensor se detendrá y abrirá sus puertas.

Cuando la cabina realice la apertura de sus puertas en su totalidad se activará un temporizador de 8 segundos cuya finalidad no es otro que simular la entrada y salida de personas y permitir que los usuarios puedan escoger su destino. Al igual que ocurriera con el anterior sistema, si el temporizador culmina o algún usuario antes de cumplirse el tiempo acciona el botón de cerrar desde el interior de la cabina, las puertas comenzarán a cerrarse.

Al ocurrir esta acción pueden presentarse dos situaciones diferentes. En la primera de ellas, un usuario puede accionar el botón de apertura de puertas desde el interior del ascensor, lo que provocará que las puertas comiencen a abrirse de nuevo, por lo que el ciclo se repetirá, es decir, apertura de puertas, temporizador de 8 segundos y cierre de puertas. Sin embargo, si las puertas se cierran en su totalidad, se producirá la activación de un temporizador de apenas 20 ms. La finalidad de este temporizador es evitar que se produzca simultáneamente el cálculo del próximo piso de destino por un lado y la continuación del ciclo del ascensor por otro. Si esto se produjera, la elección del piso podría ser errónea. Dicha elección se estudiará mediante codificación en lenguaje ST. Una vez finalizados estos 20 ms, se culmina el ciclo general volviendo al punto de partida.

Si pasados 40 segundos desde el momento en que ambos ascensores se encuentran libres no se produce ninguna llamada, el ascensor 1 adquiere una llamada para situarse en la planta baja y el ascensor 2 adquiere otra llamada para situarse en la 5ª planta, volviendo a permitir que en caso de una llamada posterior, los ascensores tarden el menor tiempo posible en llegar a cualquier planta.

Si se produjera una parada forzada de algún ascensor, esta cabina detendrá su movimiento. Si cuando se restaure el movimiento del ascensor, su llamada ha sido trasladada al otro, el ascensor que estuvo parado volverá a su planta inicial (1º planta en caso del ascensor 1, 5º planta en caso del ascensor 2). Sin embargo, si el ascensor mantiene sus llamadas, éste continuará su curso tal y como estaba realizando. Así pues, ocurre lo mismo en ambas maniobras, con la peculiaridad de que la maniobra colectiva selectiva si permitirá llamadas mientras el ascensor se recoloca en su planta origen.

El grafet general de esta secuencia de acciones del ascensor 1 aparece en la Figura 5.8. No se incluye el temporizador de 0.3 segundos ya que su función es simplemente la de asegurar que el procesador realiza sus cálculos pertinentes. Igualmente, la secuencia para el ascensor 2 es la misma variando únicamente la planta de destino en caso de traspaso de llamadas tras la parada forzada.

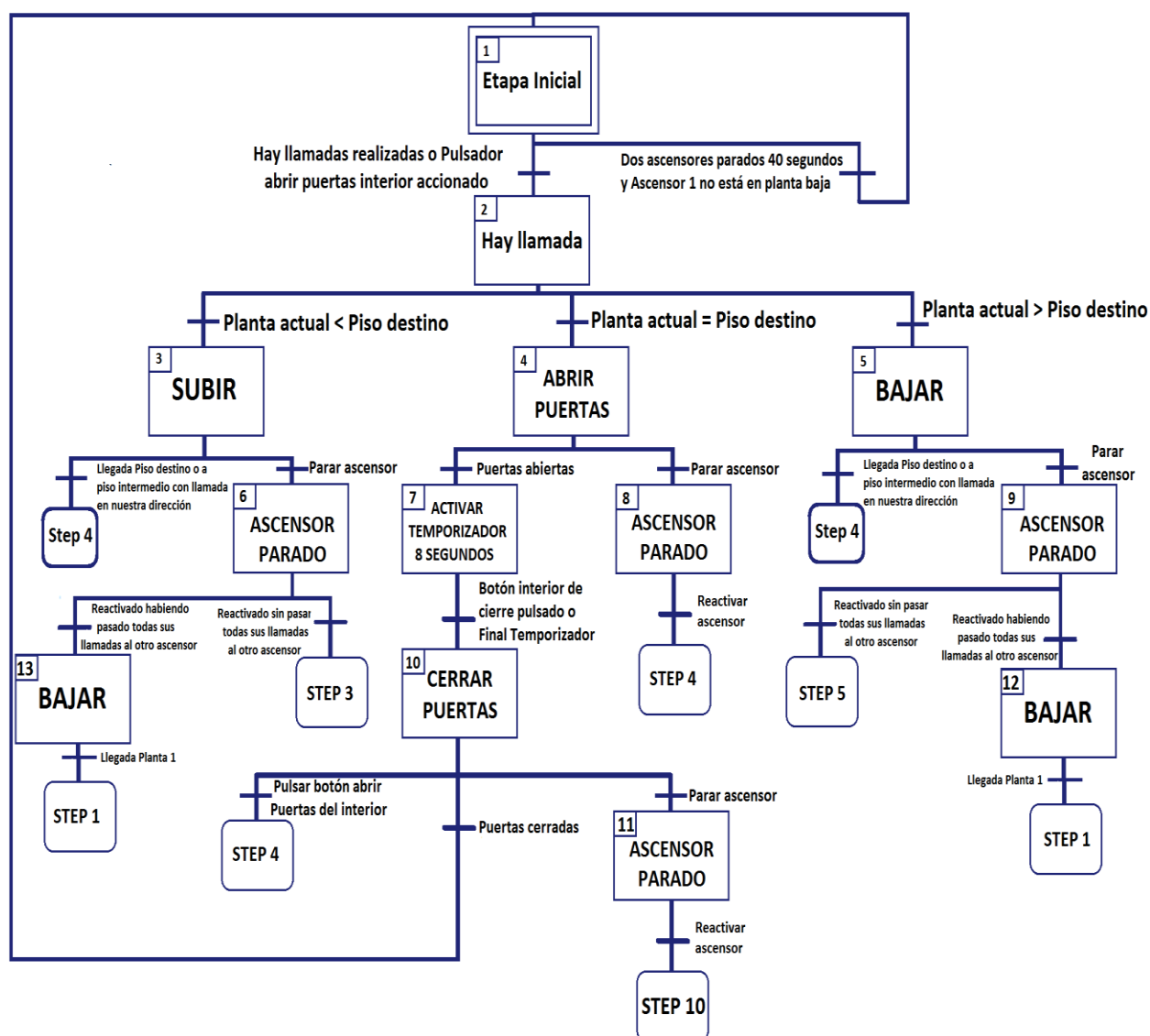


Figura 5.8 Grafet general del Ascensor 1 con maniobra colectiva selectiva

5.3.2 Lenguaje LD

La codificación incluida en este lenguaje se encarga de realizar las siguientes funciones:

- La creación del temporizador de 8 segundos que se activará cuando las puertas del ascensor se abran completamente y que al terminar activarán una señal que permitirá el cierre de las puertas.
- Al comenzar a abrirse las puertas por primera vez activará una variable que permitirá desde este momento accionar los pulsadores interiores. Esto se realiza para impedir que la simulación carezca de sentido, ya que es físicamente imposible accionar un pulsador del interior de la cabina sin encontrarse nada ni nadie dentro.
- Una acción que permite cerrar las puertas según haya terminado el temporizador de 8 segundos o se haya accionado el pulsador de cerrar puertas desde el interior.
- Un temporizador de 10 segundos, que se activa cuando uno de los dos ascensores está parado y que, si se cumple este tiempo, activa otra variable encargada de traspasar todas las llamadas del ascensor parado al otro ascensor, siempre y cuando las llamadas puedan ser realizadas por el otro ascensor.
- Siempre que un ascensor comience un movimiento, desactiva la opción de traspaso para evitar mantenerse activo indefinidamente en caso de haberse realizado un traspaso con anterioridad.
- Existe una variable que se encuentra activada indefinidamente que permite comprobar en todo momento el estado de ocupación de cada ascensor. Esta comprobación se realiza mediante lenguaje ST.
- Cada vez que un ascensor inicia un movimiento, activa una variable en función de si el desplazamiento realizado es de subida o de bajada, con el objetivo de recordar, en el momento en que se detenga, cuál fue su último movimiento y poder mantenerlo en caso de ser posible, tal y como se estipula en la maniobra colectiva selectiva.
- Si alguna puerta se abre en su totalidad, se activa la variable general de puertas abiertas necesaria para que el graficet siga su curso.
- Al cerrar totalmente las puertas, se activará una variable que permitirá realizar un estudio de la próxima planta de destino en caso de haber llegado a dicha planta, el cual también estará recogido en lenguaje ST.

5.3.3 Lenguaje ST

Igual que ocurriera con la simulación para la maniobra universal, el lenguaje ST es el encargado de realizar la programación de todos los casos posibles que se pueden dar a la hora de realizar una llamada del ascensor en función del momento en que se produzca. Así, este lenguaje incluye las siguientes casuísticas:

- Si ambos ascensores están disponibles y vacíos, la petición será atendida por el ascensor que se encuentre más cerca de la planta de destino siempre y cuando se encuentre detenido, o por aquel que tenga la planta de reposo más cercana al piso de destino si ambas cabinas se están recolocando en dichas plantas de reposo. Si no se cumplen ninguna de estas circunstancias, la atenderá el otro ascensor.
- Si un ascensor está libre y el otro está ocupado, la llamada se le asignará únicamente a aquel que se encuentre vacío. Existe la excepción de que, si el ascensor vacío está realizando alguna acción, como la apertura o cierre de puertas, la petición también será recogida por el otro ascensor por si existe la posibilidad de ser atendida con mayor rapidez.
- Si ambos ascensores están ocupados y se produce una nueva petición, las llamadas se le asignarán a los dos.
- Si los dos ascensores se encuentran en parada forzada, independientemente del tipo de llamadas en espera que tuvieran, todas las llamadas se perderán y volverán a sus plantas de reposo.
- Si se produce una llamada interior en un ascensor, si el piso accionado es inferior a la planta actual, se registrará en memoria como petición de bajada; en cambio, si el piso pulsado es superior a la planta actual, se guardará como llamada de subida.
- En caso de atenderse la llamada del piso de destino registrado por el ascensor, se pueden dar 4 casos diferentes en función de la llamada que se haya atendido y del último desplazamiento realizado por la cabina para la nueva asignación del piso de destino:

- Si el movimiento para atender al piso de destino fue de subida y no era una llamada que pretendía bajar, para encontrar un nuevo piso de destino, al cerrar las puertas se comprueban las llamadas para subir de los pisos superiores al actual. En caso de no existir, se busca la planta más alta registrada para bajar. En caso de que tampoco exista ninguna, el controlador buscará la planta más baja que desee descender de planta. Por último, si no se encuentra ninguna llamada de este estilo significará que no hay llamadas designadas, por lo que el controlador no tiene registrada ninguna llamada.
- Si el movimiento para atender al piso de destino fue de bajada y no era una llamada que pretendía subir, para encontrar un nuevo piso de destino, al cerrar las puertas se comprueban las llamadas para bajar de los pisos inferiores al actual. En caso de no encontrar ninguna, se busca la planta más baja registrada para subir. Si tampoco existe ninguna, se comprueba la planta más alta que desee ascender de planta. Si volvemos a fracasar significará que el ascensor no tiene en ese momento ninguna llamada pendiente.
- Si el movimiento para atender al piso de destino fue de bajada y era una llamada que pretendía subir, para encontrar un nuevo piso de destino, se analizan las llamadas para subir de los pisos superiores al actual, buscando así mantener el movimiento de la llamada atendida. Si el análisis no da resultados, el siguiente paso, por coherencia, es mantener el último movimiento del ascensor el mayor tiempo posible. Así pues, primeramente se buscará entre las plantas inferiores a la de la planta actual con el objetivo de encontrar una llamada realizada para bajar. Si esta llamada no se encuentra, se procede a analizar las plantas que se encuentran por debajo de la actual con llamada realizada para ascender. Si se sigue sin encontrar resultado, se comprueban las llamadas de plantas superiores para descender. En caso de que el resultado de la búsqueda sea negativo significará que la memoria del ascensor no tiene ninguna llamada registrada.
- Si el movimiento para atender al piso de destino fue de subida y era una llamada que pretendía bajar, en primer lugar se analizarán las llamadas registradas para bajar de los pisos inferiores al actual, con el objetivo de mantener el movimiento de la llamada a la cual se ha atendido el mayor tiempo posible. En caso de no encontrarla, se intentará mantener la dirección del desplazamiento último del ascensor, es decir, primero se analizará las plantas superiores a la actual para encontrar una llamada realizada para subir y, en caso de no encontrarla, se buscará entre las plantas mayores a la actual que desee descender. Si se continúa la dinámica de no encontrar llamada de este tipo, tan solo queda comprobar las llamadas de pisos inferiores realizadas por usuarios que desean ascender. Si no se obtiene resultado será porque el ascensor se encuentra sin llamadas pendientes.

- Cuando un ascensor se detiene en una planta para atender una petición de llamada, las luces del interior del ascensor y de la botonera exterior de pasillo de este piso se apagarán, y la memoria de este ascensor eliminará esta llamada de su registro. Si la planta objetivo del otro ascensor no es la misma, este ascensor también la eliminará de su memoria, siempre y cuando un usuario del interior de la cabina no haya accionado el botón de ese piso. También se obviará la llamada si, aunque tenga esa planta como destino, existe alguna petición de usuario que pueda ser resuelta sin variar el transcurso del movimiento de este segundo ascensor o si se encuentra detenido en ese instante.

- Una cabina se detendrá en un piso que no sea su destino si encuentra una llamada en su trayecto que ha sido realizada con la misma dirección del transcurso de la cabina, a menos que el ascensor esté realizando un movimiento contrario al de la llamada atendida, es decir, no esté ascendiendo para atender una petición de bajada o descendiendo para atender una petición de subida. En ese caso, no se detendrá.

- Se produce un constante análisis del número de peticiones pendientes para conocer si el ascensor se encuentra con alguna llamada en su registro o si por el contrario está libre de llamadas.

- Cuando un ascensor se encuentre detenido de manera forzada durante más de 10 segundos, todas las llamadas pendientes de este ascensor que hayan sido realizadas desde la botonera de pasillo, pasarán automáticamente a la memoria del otro ascensor y eliminándose del registro del primero, con el fin de poder ser realizada con la mayor antelación posible. Además, todas las llamadas que se realicen durante el tiempo en que se encuentre ese ascensor detenido, serán recogidas exclusivamente por el ascensor activo.

- Si trascurren 40 segundos desde la última llamada atendida por alguno de los dos ascensores, si el ascensor 1 no se encuentra en la planta 1, éste adquiere una llamada para trasladarse a dicha planta. Lo mismo ocurrirá con el ascensor 2 si no se encuentra en la planta 5. Si en el trayecto a dichas posiciones alguno de los ascensores adquiere una petición de llamada que por sus características puede realizar, éste la ejecutará y al acabar de

hacerlo, la llamada adquirida a la planta de traslado desaparece. En cambio, si el ascensor adquiere una llamada que por sus condiciones no puede ser atendida antes de llegar a la planta de reposo, primero asistirá a la planta donde ha de trasladarse y después atenderá la petición pendiente.

Además de todos estos casos, igual que ocurre con la simulación de la otra maniobra, existe una POU en lenguaje ST que se encarga de realizar los movimientos de las puertas, de mover la cabina en ambas direcciones, de reconocer el nivel en el cual se encuentra la cabina, etc. Todo esto, será apoyado con una visualización de ambos ascensores, la cual será igual a la de la maniobra universal con la única diferencia de que en esta simulación se tienen dos botones exteriores de pasillo que permitirán escoger la dirección deseada por el usuario, uno para subir y otro para bajar, exceptuando los pisos extremos.

Destacar, por último, las variables globales que permiten el correcto funcionamiento de esta maniobra a las cuales se les ha agregado una breve descripción para conocer su labor dentro de la codificación.

0001	VAR_GLOBAL	
0002		
0003	(*ASCENSOR 1 Y 2*)	
0004		
0005	PULSADOR_SUB: ARRAY [1..10] OF BOOL;	(*Pulsadores de cada planta exterior de ambos ascensores*)
0006	PULSADOR_BAJ: ARRAY [1..10] OF BOOL;	(*Pulsadores de cada planta exterior de ambos ascensores*)
0007	LUZ_SUB: ARRAY[1..10] OF BOOL;	(*Luz de los pulsadores exteriores*)
0008	LUZ_BAJ: ARRAY[1..10] OF BOOL;	(*Luz de los pulsadores exteriores*)
0009	FT10: BOOL;	(*Fin temporizador de 10 segundos*)
0010	AMBOS_VACIOS: BOOL;	(*Se activa cuando ambos ascensores están vacíos y han pasado 40 segundos*)
0011	SIEMPRE: BOOL;	(*Señal activa en todo momento*)
0012	START: BOOL:=TRUE;	(*Permite que, al comenzar, ambos ascensores ocupen sus plantas de reposo*)
0013		
0014		
0015	(*ASCENSOR 1*)	
0016		
0017	AT8: BOOL;	(*Inicio Temporizador 8 segundos para mantener las puertas abiertas*)
0018	FT8: BOOL;	(*Fin Temporizador 8 segundos para mantener las puertas abiertas*)
0019	T0:BOOL;	(*Inicio tiempo 0.3 seg*)
0020	FT0: BOOL;	(*Final temporizador 0.3 seg*)
0021	FLAG: BOOL;	(*Permite accionar un pulsador interior cuando se abran las puertas por 1ª vez*)
0022		
0023	IGUAL: BOOL;	(*Indica que la posición de la cabina es igual al destino*)
0024	MAYOR: BOOL;	(*Indica que la posición de la cabina es mayor al destino*)
0025	MENOR: BOOL;	(*Indica que la posición de la cabina es menor al destino*)
0026		
0027	PARADO: BOOL;	(*El ascensor se encuentra parado*)
0028	BAJAR: BOOL;	(*Al activarse hace que el ascensor 1 comience a bajar*)
0029	SUBIR: BOOL;	(*Al activarse hace que el ascensor 1 comience a subir*)
0030	Vengo_de_abajo: BOOL;	(*Indica el último movimiento realizado (subir)*)
0031	Vengo_de_arriba: BOOL;	(*Indica el último movimiento realizado (bajar)*)
0032	Recordar_subir: BOOL;	(*Recuerda el movimiento realizado para cuando se detenga (subir)*)
0033	Recordar_bajar: BOOL;	(*Recuerda el movimiento realizado para cuando se detenga (bajar)*)
0034	RECOGIDA_BAJAR: BOOL;	(*Indica que el ascensor sube para atender una petición de bajada*)
0035	RECOGIDA_SUBIR: BOOL;	(*Indica que el ascensor baja para atender una petición de subida*)
0036	STOP1: BOOL;	(*Estado de parada forzada*)
0037	TRASPASO: BOOL;	(*Se activa cuando el ascensor cede sus llamadas al otro*)
0038	RECOLOCA: BOOL;	(*Indica que el ascensor debe situarse en el piso de reposo*)
0039		
0040	SENSORES: BOOL;	(*Se activa cuando cualquiera de los sensores se encuentra activo*)
0041	SENSOR: ARRAY [1..10] OF BOOL;	(*Sensores de cada planta del ascensor*)
0042	PULSADOR_INT1: ARRAY [1..10] OF BOOL;	(*Pulsadores interiores del ascensor*)
0043	LUZ_INT: ARRAY[1..10] OF BOOL;	(*Luz de los pulsadores interiores*)
0044	PCP: BOOL;	(*Pulsador cerrar puerta*)
0045	PAP: BOOL;	(*Pulsador abrir puerta*)

Figura 5.9 Variables globales utilizadas en la simulación de los ascensores la maniobra colectiva selectiva (1)

0046		
0047	OCUPADO: BOOL;	(*Indica que el ascensor aún tiene peticiones pendiente*)
0048	AUXILIAR: INT;	(*Variable de apoyo para la comprobación de PISO_PULSADO*)
0049	PISO_PULSADO: INT;	(*Indica el piso de destino*)
0050	TURN0_SUB: ARRAY [1..10] OF BOOL;	(*Indica aquellos pisos de subida que se deben atender*)
0051	TURN0_BAJ: ARRAY [1..10] OF BOOL;	(*Indica aquellos pisos de bajada que se deben atender*)
0052		
0053	PUERTA_DER: ARRAY [1..10] OF LREAL;	(*Movimiento de la puerta derecha de cada planta*)
0054	PUERTA_IZQ: ARRAY [1..10] OF LREAL;	(*Movimiento de la puerta izquierda de cada planta*)
0055	MAX_APE: ARRAY [1..10] OF BOOL;	(*Maxima apertura de cada puerta*)
0056	MAX_CIE: ARRAY [1..10] OF BOOL;	(*Maximo cierre de cada puerta*)
0057	MAX_CIER: BOOL;	(*Se activa cuando las puertas de todos los rellanos están cerradas*)
0058	COND_CERRAR: BOOL;	(*Indica que se ha producido una de las condiciones que permite cerrar las puertas*)
0059	MAXIMA_APERTURA: BOOL;	(*Se activa cuando las puertas del piso en el que nos hallemos está abierta*)
0060	ABRIENDO_PUERTAS: BOOL;	(*Estado del ascensor mientras abre las puertas*)
0061	CERRANDO_PUERTAS: BOOL;	(*Estado del ascensor mientras cierra las puertas*)
0062	ACABA_CERRAR: BOOL;	(*Se activa justo cuando las puertas se cierran*)
0063		
0064	DISPLAY1: INT;	(*Permite conocer el piso en el que la cabina se encuentra *)
0065	PLANTA: INT;	(*Sirve para registrar el piso actual*)
0066	ABRE: BOOL;	(*Indica la llegada a una planta en la cual puede detenerse*)
0067	VAR1: BOOL;	(*Variable de apoyo auxiliar*)
0068	NO_ACEPTAMOS: BOOL;	(*El ascensor está parado pero realizando alguna acción del grafcet*)
0069		
0070		
0071	(*ASCENSOR 2*)	
0072		
0073	AT82: BOOL;	(*Inicio Temporizador 8 segundos para mantener las puertas abiertas*)
0074	FT82: BOOL;	(*Fin Temporizador 8 segundos para mantener las puertas abiertas*)
0075	T00: BOOL;	(*Inicio tiempo 0.3 seg*)
0076	FT00: BOOL;	(*Final temporizador 0.3 seg*)
0077	FLAG2: BOOL;	(*Permite accionar un pulsador interior cuando se abran las puertas por 1ª vez*)
0078		
0079	IGUAL2: BOOL;	(*Indica que la posición de la cabina es igual al destino*)
0080	MAJOR2: BOOL;	(*Indica que la posición de la cabina es mayor al destino*)
0081	MEJOR2: BOOL;	(*Indica que la posición de la cabina es menor al destino*)
0082		
0083	PARADO2: BOOL;	(*El ascensor se encuentra parado*)
0084	Subir2: BOOL;	(*Al activarse hace que el ascensor 2 comience a subir*)
0085	Bajar2: BOOL;	(*Al activarse hace que el ascensor 2 comience a bajar*)
0086	Vengo_de_abajo2: BOOL;	(*Indica el último movimiento realizado (subir)*)
0087	Vengo_de_arriba2: BOOL;	(*Indica el último movimiento realizado (bajar)*)
0088	Recordar_subir2: BOOL;	(*Recuerda el movimiento realizado para cuando se detenga (subir)*)
0089	Recordar_bajar2: BOOL;	(*Recuerda el movimiento realizado para cuando se detenga (bajar)*)
0090	RECOGIDA_BAJAR2: BOOL;	(*Indica que el ascensor sube para atender una petición de bajada*)
0091	RECOGIDA_SUBIR2: BOOL;	(*Indica que el ascensor baja para atender una petición de subida*)
0092	STOP2: BOOL;	(*Estado de parada forzada*)
0093	TRASPASO2: BOOL;	(*Se activa cuando el ascensor cede sus llamadas al otro*)
0094	RECOLOCA2: BOOL;	(*Indica que el ascensor debe situarse en el piso de reposo*)
0095		
0096	SENSORES2: BOOL;	(*Se activa cuando cualquiera de los sensores se encuentra activo*)
0097	SENSOR2: ARRAY [1..10] OF BOOL;	(*Sensores de cada planta del ascensor*)
0098	PULSADOR_INT2: ARRAY [1..10] OF BOOL;	(*Pulsadores interiores del ascensor*)
0099	LUZ_INT2: ARRAY [1..10] OF BOOL;	(*Luz de los pulsadores interiores*)
0100	PCP2: BOOL;	(*Pulsador cerrar puerta*)
0101	PAP2: BOOL;	(*Pulsador abrir puerta*)
0102		
0103	OCUPADO2: BOOL;	(*Indica que el ascensor aún tiene peticiones pendiente*)
0104	AUXILIAR2: INT;	(*Variable de apoyo para la comprobación de PISO_PULSADO2*)
0105	PISO_PULSADO2: INT;	(*Indica el piso de destino*)
0106	TURN02_SUB: ARRAY [1..10] OF BOOL;	(*Indica aquellos pisos de subida que se deben atender*)
0107	TURN02_BAJ: ARRAY [1..10] OF BOOL;	(*Indica aquellos pisos de bajada que se deben atender*)
0108		
0109	PUERTA_DER2: ARRAY [1..10] OF LREAL;	(*Movimiento de la puerta derecha de cada planta*)
0110	PUERTA_IZQ2: ARRAY [1..10] OF LREAL;	(*Movimiento de la puerta izquierda de cada planta*)
0111	MAX_APE2: ARRAY [1..10] OF BOOL;	(*Maxima apertura de cada puerta*)
0112	MAX_CIE2: ARRAY [1..10] OF BOOL;	(*Maximo cierre de cada puerta*)
0113	MAX_CIER2: BOOL;	(*Se activa cuando las puertas de todos los rellanos están cerradas*)
0114	MAXIMA_APERTURA2: BOOL;	(*Se activa cuando las puertas del piso en el que nos hallemos está abierta*)
0115	COND_CERRAR2: BOOL;	(*Indica que se ha producido una de las condiciones que permite cerrar las puertas*)
0116	ABRIENDO_PUERTAS2: BOOL;	(*Estado del ascensor mientras abre las puertas*)
0117	CERRANDO_PUERTAS2: BOOL;	(*Estado del ascensor mientras cierra las puertas*)
0118	ACABA_CERRAR2: BOOL;	(*Se activa justo cuando las puertas se cierran*)
0119		
0120	DISPLAY2: INT;	(*Permite conocer el piso en el que la cabina se encuentra*)
0121	PLANTA2: INT;	(*Sirve para registrar el piso actual*)
0122	ABRE2: BOOL;	(*Indica la llegada a una planta en la cual puede detenerse*)
0123	VAR2: BOOL;	(*Variable de apoyo auxiliar*)
0124	NO_ACEPTAMOS2: BOOL;	(*El ascensor está parado pero realizando alguna acción del grafcet*)
0125		
0126	END_VAR	

Figura 5.10 Variables globales utilizadas en la simulación de los ascensores la maniobra colectiva selectiva (2)

6 UNITY PRO

Unity Pro es un software creado por Schneider Electric que, al igual que ocurre con CoDeSys, nos ofrece la posibilidad de programar, depurar y operar con autómatas de las gamas Modicom Premium, Atrium y Quantum PLC. Debido a que el autómata utilizado para la ejecución de este proyecto es el Modicom M340 de Telemecanique, perteneciente a Schneider Electric, es necesario la utilización de este software informático para poder ponerlo en funcionamiento.

Su aplicación es parecida a la de CoDeSys, ya que permite ser utilizado para el control de infraestructuras, el control de fabricación o de procesamiento de lotes [29].

Utiliza los cinco lenguajes del IEC 61131-1 lo que permite el uso de todas las herramientas de depuración y diagnóstico más accesibles aumentando así la productividad de desarrollo y la facilidad de ejecución.

Los requisitos mínimos para la utilización de Unity Pro se encuentran reflejados en la siguiente tabla.

	MINIMO	RECOMENDADO
Sistema	Pentium 800 MHz	1.2 GHz
Memoria RAM	256 MB	512 MB
Disco Duro	2 GB	4 GB
Sistema Operativo	Windows 2000 o superior	
Pantalla	SVGA o mayor resolución	
Periféricos	Ratón y teclado	
Internet	Solución idónea para registrarse	

Tabla 6.1 Características básicas de Unity Pro [30]

6.1 Introducción al Software

En este apartado se va a realizar una introducción al programa a la vez que se explican los pasos necesarios que se han llevado a cabo para la configuración del autómata y la creación de las diferentes POUs que conforman el programa que permite el manejo de la maqueta de ascensores.

Al iniciar el software observamos la ventana de explotación de la Figura 6.2, en la que se incluye:

- **Barra de menús:** Permite realizar las distintas funciones.
- **Barra de iconos:** Incluye accesos a las funciones principales del programa.
- **Barra de instrumentos:** Aparecen las herramientas más utilizadas dentro del lenguaje en el que se encuentre.
- **Barra de estado:** Indicará si el programa se encuentra ONLINE/OFFLINE, en estado de simulación, RUNNING, etc.
- **Panel de resultados:** Contiene los errores que puedan aparecer en la codificación. Para ello, debe realizarse un previo análisis del código creado.

- **Explorador de proyectos:** es el eje del programa e incluye tanto la información del PLC creado como las variables, los bloques funcionales o las distintas POU's, entre otros.
- **Editor del programa:** Aquí se encontrarán las ventanas que se tengan abiertas, ya sean POU's, variables elementales, tablas de animación, etc.

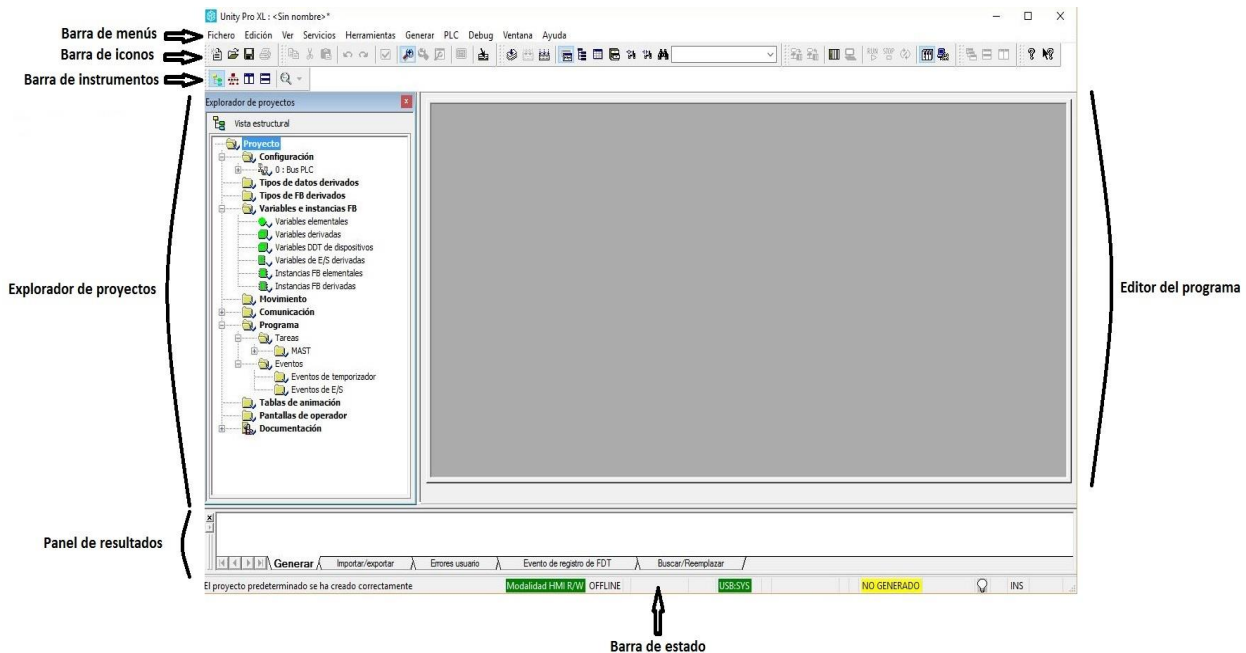


Figura 6.1 Ventana principal del programa Unity Pro

En primer lugar, se realiza la creación de un archivo nuevo. Para ello, en la barra de menús, se realiza la acción **Fichero → Nuevo**. Una vez realizado esto, aparecerá una ventana en la que se debe escoger de entre una serie de opciones el PLC y el bastidor que se desee utilizar. En este proyecto se utilizará como PLC el Modicon M340 BMX P34 2020 y como bastidor el BMX XBP 0400 que contiene 4 SLOTS y hacemos clic en **Aceptar**.

Una vez escogido esto, aparecerá una ventana como la de la Figura 6.2, en la cual escogeremos en el BUS PLC la versión BMX P34 2020 02.00.

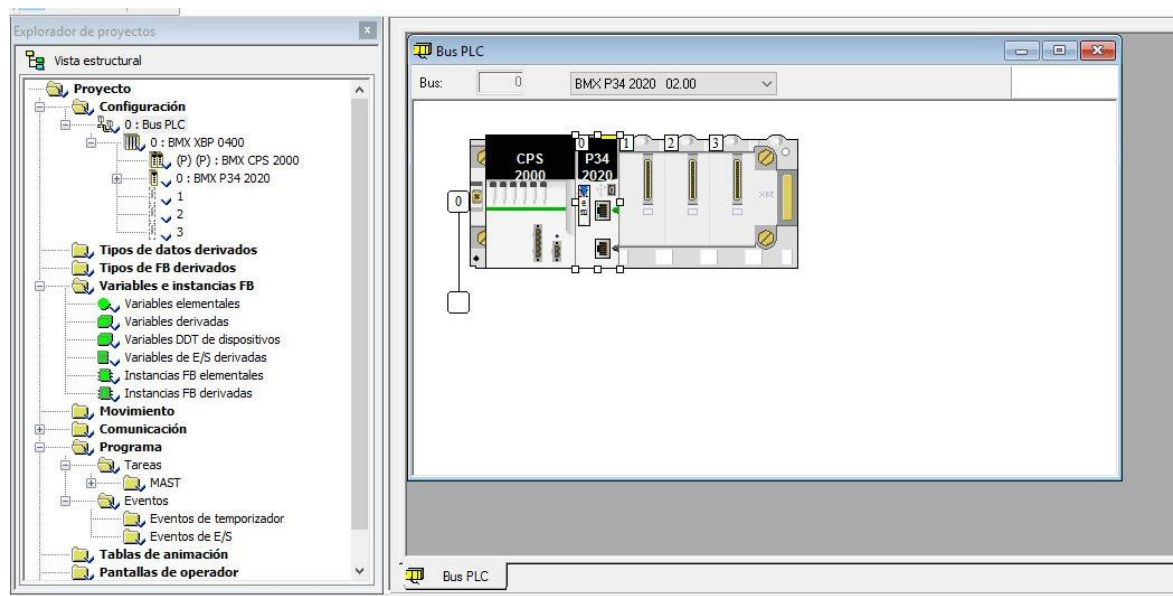


Figura 6.2 Ventana del BUS PLC utilizado sin la elección de los módulos de E/S

A continuación, añadiremos los módulos que se utilicen en el autómatas. En este proyecto se recuerda que estos módulos son el BMX DDI 3202K y el BMX DDM 16022. Para introducirlos dentro del bastidor del programa, hacemos doble clic en la primera ranura y buscamos el primer módulo. Es un módulo binario por lo que buscamos entre estos y hacemos clic en **Aceptar**.

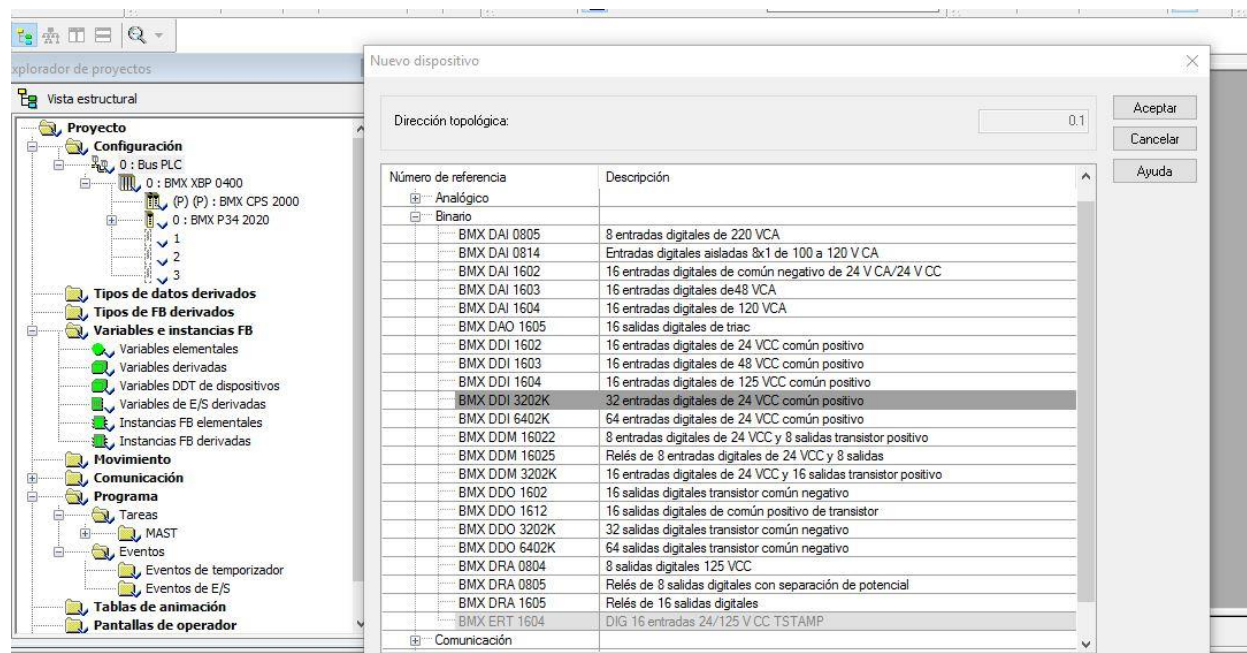


Figura 6.3 Menú de selección de módulos de E/S

Para el segundo módulo se realizará la misma acción pero haciendo doble clic en la segunda ranura, ya que la primera ya ha sido ocupada. Así, ya se encuentran instalados y listos para ser utilizados los módulos de los que el usuario dispone quedando reflejado de la siguiente manera.

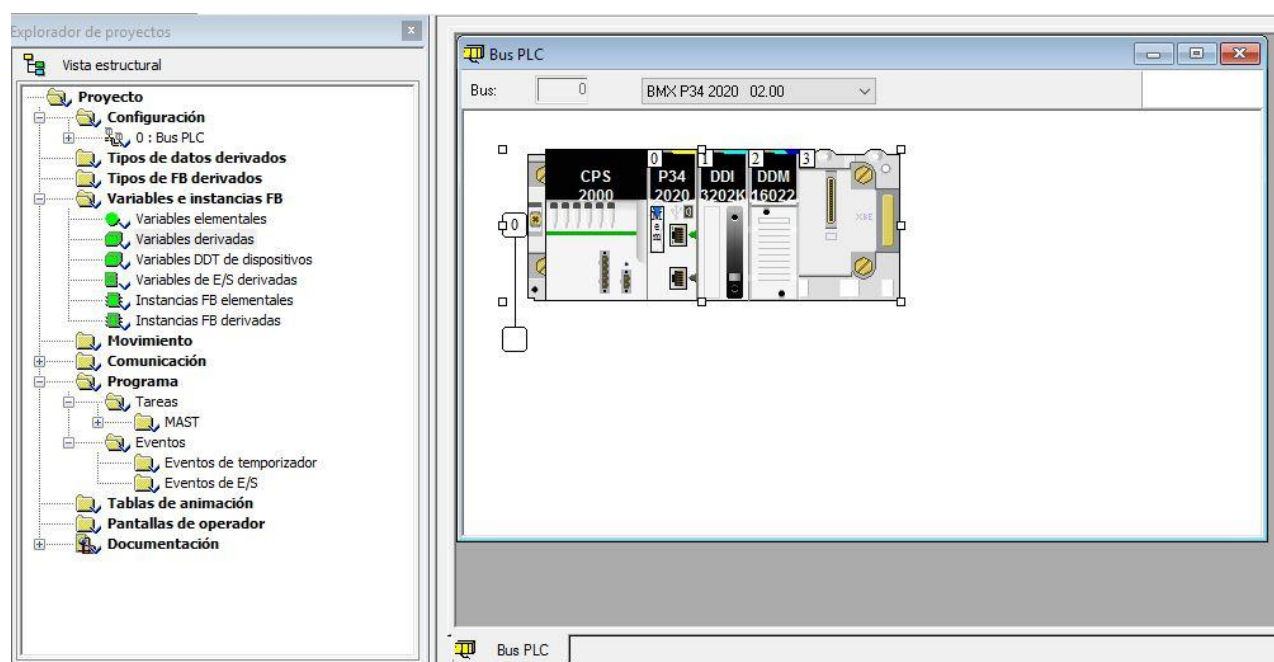


Figura 6.4 Ventana del BUS PLC con la elección de los módulos de E/S realizadas

El siguiente paso es la creación de las diferentes variables que se van a utilizar en el proyecto. Para ello, en el **Explorador de proyectos** hacemos clic en **Variables elementales** y comenzamos a rellenar el nombre y el tipo de variable (BOOL, EBOOL, ARRAY...) de cada una de ellas. Además, aquellas variables que tengan relacionadas una entrada o salida deben asociarse escribiendo el nombre de su entrada o salida correspondiente.

Tras la declaración de variables, lo siguiente es crear los diferentes códigos que se encargarán de que la simulación siga su curso. Para su creación se debe hacer clic, dentro del **Explorador de proyectos** en **Programas**→**Tareas**→**MAST**. En MAST hacer clic con el botón derecho sobre **Secciones** y elegir la opción **Nueva Sección**. Tras realizarse esto, aparecerá la siguiente ventana.

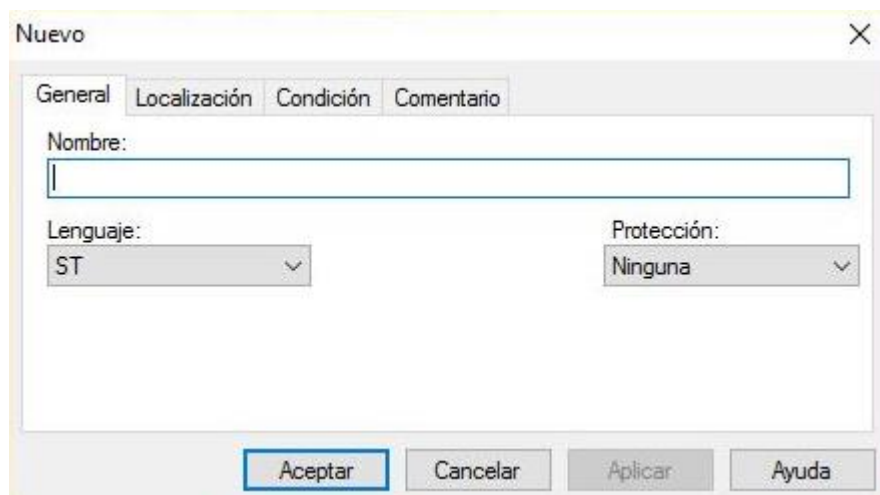


Figura 6.5 Menú de configuración de una POU

En ella escogeremos el nombre de nuestra POU, el lenguaje que utilizará y si se desea añadir alguna protección.

Además de todo esto, es posible la creación de una pantalla de operador para la recreación visual de los elementos codificados ya que, además de incluir un modo estándar que permite transferir el proyecto a un PLC real, también es posible utilizarlo en un estado de simulación, para el cual esta pantalla es bastante idónea.

Para su creación, basta con hacer clic con el botón derecho en **Pantalla de operador** y seleccionar **Nueva pantalla**. Tras esta acción, se escoge un nombre para la nueva pantalla. Con esto, la creación de la pantalla está realizada y el usuario puede crear las figuras e imágenes que crea oportunas utilizando la barra de herramientas.

También es posible la creación de tablas de animación en las cuales el usuario podrá observar como cambian de estado las variables que ha creado, y ha introducido en dicha tabla, para tener controlado el estado de la simulación en todo momento. Para crearla basta con hacer clic con el botón derecho en **Tabla de animación** y seleccionar la opción **Nueva Tabla**, se le asigna un nombre y hacemos clic en **Aceptar**. Así, el usuario tan solo debe incluir las variables que estime oportunas.

Por último, una vez creadas las diferentes POUs, para que pueda ser reconocido por el PLC, primero el código debe ser analizado. Para ello hacemos clic en el Menú **Generar** → **Analizar proyecto**. Tras ser analizado y observar que no hay errores, el siguiente paso es generar los cambios en el proyecto, por lo que se hace clic de nuevo en el menú **Generar** → **Regenerar todo el proyecto**, para así tener todo el proyecto actualizado.

Toda vez generado el proyecto, el tercer paso es conectarlo al PLC. Para ello, primero se debe tener marcada la opción **Modalidad estándar** y tras esto, hacer clic en el menú **PLC** → **Conectar**.

Tras su conexión se debe transferir el proyecto al PLC. Así pues, se hace clic en el menú **PLC** → **Transferir proyecto al PLC**. Por último, tan solo queda ejecutar el programa. Para ello, se hace clic en el menú **PLC** → **Ejecutar**.

6.2 Maqueta de dos ascensores de 4 plantas con maniobra universal

El programa que se utilizará para que la maqueta realice la maniobra universal contiene, al igual que ocurría con la simulación, tres lenguajes diferentes de codificación: Lenguajes SFC, LD y ST.

La programación utilizada es, en líneas generales, la misma que la de la simulación ya que es igualmente válida para ser simulado y para ser utilizado en una maqueta real. Sin embargo, se han realizado una serie de cambios para adecuarlo a la maqueta, ya que no se dispone de los mismos elementos que los utilizados en la simulación.

Los cambios realizados son los siguientes:

- **Lenguaje SFC**

- La planta de reposo del ascensor 2 en lugar de ser la planta 5 (inexistente) será la planta 3.
- En la maqueta no existen puertas, por lo que los pasos de apertura y cierre de puertas no están definidos. Una vez llegados al piso destino se producirá una espera de 5 segundos que permitiría la entrada y/o salida de los usuarios y tras esto, una espera extra de 3 segundos para permitir seleccionar un botón interior.
- Se elimina la posibilidad de accionar el pulsador de la opción de abrir o cerrar las puertas desde el interior con sus consecuentes acciones ya que no existen puertas.
- La planta 1 de la simulación se corresponde ahora con la Planta Baja.

- **Lenguaje LD**

- Al no existir puertas, las condiciones de cerrar de ambos ascensores se suprimen.
- Se permite la pulsación de un botón interior justo cuando comienza el temporizador de 5 segundos en lugar de realizarse cuando comienzan a abrirse las puertas.
- Se añaden variables nuevas debido a que Unity Pro no permite asociar a un mismo sensor del lenguaje SFC dos condiciones a la vez, por lo que dichas condiciones se incluyen en este lenguaje activando así otra variable.
- Se añaden entradas y salidas reales del autómatas que deben ser activadas cuando se produzca alguna condición y que en la simulación no eran necesarias.

- **Lenguaje ST**

- Desaparece toda aquella codificación relacionada con elementos de simulación como fuera simular la subida o bajada del ascensor, apertura o cierre de puertas, etc., ya que no será necesario simularlo al disponer de la maqueta.
- Las luces internas de la botonera de las cabinas no aparecen ya que no se dispone de leds que hagan esta función, por lo que no será necesario ni encenderlas ni apagarlas en ningún momento.
- Se adapta la codificación al estudio de los casos para un ascensor de 4 plantas.

- **Variables elementales**

- Desaparecen todas las variables relacionadas con la apertura y cierre de puertas de ambos ascensores.
- No se incluyen tampoco las variables relacionadas con las luces interiores.
- No se incluyen variables relacionadas con la simulación, como por ejemplo, DISPLAY.
- Aparecen variables nuevas que se utilizarán para las condiciones que no fueron posibles incluir en los sensores del lenguaje SFC y otras variables relacionadas con la parada del ascensor.

VARIABLE	DESCRIPCIÓN
AMBOS_LIBRES	Se activa al estar activo AMBOS_VACIOS y el ascensor 1 no se encuentra en la planta de reposo
AMBOS_LIBRES2	Se activa al estar activo AMBOS_VACIOS y el ascensor 2 no se encuentra en la planta de reposo
AMBOS_VACIOS	Variable que se activa cuando ambos ascensores están vacíos
PARA_ASC1	Relacionada con el boton S de la cabina 1
PARA_ASC2	Relacionada con el boton S de la cabina 2
REACTIV_ASC1	Relacionada con el boton A de la cabina 1
REACTIV_ASC2	Relacionada con el boton A de la cabina 2

Tabla 6.2 Variables diferentes o añadidas respecto a la simulación de la maniobra universal

6.3 Maqueta de dos ascensores de 4 plantas con maniobra colectiva selectiva

Para esta maniobra se vuelve a contar de nuevo con la codificación en los 3 lenguajes que se han utilizado con anterioridad: SFC, LD y ST.

Igualmente, como ocurriera con la maniobra universal, el funcionamiento de la maqueta con maniobra colectiva selectiva es, por lo general, parecido al utilizado para la simulación de esta maniobra, aunque también ha sido necesario realizar una serie de cambios, con respecto a la simulación, y eliminar o añadir elementos para permitir adaptarlo a la maqueta de la que se dispone. Estos cambios son:

- **Lenguaje SFC**
 - El piso de reposo del ascensor 2 pasa de ser la planta 5 a la planta 3.
 - Al no existir puertas en la maqueta, los pasos de apertura y cierre de puertas desaparecen, por lo que al llegar un ascensor al piso de destino, se produciría el comienzo del temporizador, que en este caso será de 8 segundos.
 - Desaparece también la opción de realizar una determinada acción al pulsar el botón de abrir o cerrar las puertas desde el interior, debido a que, como se ha comentado, no existen puertas.
 - La planta 1 en la simulación equivale ahora a la Planta 0 o Baja.
- **Lenguaje LD**
 - Se eliminan todas las acciones relacionadas con la apertura y cierre de puertas.
 - Se permite la primera llamada desde el interior de la cabina en el momento en que un ascensor activa por primera vez el temporizador de 8 segundos, el cual significará que se ha realizado la primera parada.
 - Se produce la activación de variables nuevas al no permitirse introducir más de una condición en los sensores del lenguaje SFC, por lo que son incluidas en este tipo de codificación.
 - Se crea un nuevo archivo en este lenguaje que relaciona de variables con entradas o salidas reales del autómatas que permitirán su activación.
- **Lenguaje ST**
 - Se elimina toda aquella codificación que tenga como función simular el movimiento de cada una de las partes del ascensor debido a que se trabaja con un sistema real.
 - Se adapta toda la codificación al análisis para ascensores de 4 plantas, incluida la Planta Baja.

- **Variables elementales**

- Se excluyen todas las variables destinadas a la apertura y cierre de puertas de ambos ascensores.
- Las variables de luz interior adquiere otra funcionalidad.
- Se eliminan variables utilizadas para la representación visual de los ascensores.
- Aparecen nuevas variables debido a la imposibilidad de introducir varias condiciones en los sensores del lenguaje SFC y otras variables relacionadas con la parada del ascensor.

VARIABLE	DESCRIPCIÓN
AMBOS_LIBRES	Se activa al estar activo AMBOS_VACIOS y el ascensor 1 no se encuentra en la planta de reposo
AMBOS_LIBRES2	Se activa al estar activo AMBOS_VACIOS y el ascensor 2 no se encuentra en la planta de reposo
AMBOS_VACIOS	Variable que se activa cuando ambos ascensores están vacíos
ABRIMOS	Se activa cuando la cabina 1 llega al piso destino o cuando se activa ABRE
ABRIMOS2	Se activa cuando la cabina 2 llega al piso destino o cuando se activa ABRE2
ACTUALIZA	Apaga las luces y comprueba el estado del otro ascensor al detenerse
ACTUALIZA2	Apaga las luces y comprueba el estado del otro ascensor al detenerse
LIBRE	Se activa al estar activo AMBOS_VACIOS y el ascensor 1 no se encuentra en la planta de reposo
LIBRE2	Se activa al estar activo AMBOS_VACIOS y el ascensor 2 no se encuentra en la planta de reposo
OCUP	Tras una parada forzada del ascensor 1, permite mantener sus llamadas
OCUP2	Tras una parada forzada del ascensor 2, permite mantener sus llamadas
NO_OCUP	Tras una parada forzada del ascensor 1, impide mantener sus llamadas
NO_OCUP2	Tras una parada forzada del ascensor 2, impide mantener sus llamadas
LUZ_INT1	Permite conocer si una llamada ha sido realizada desde el interior del asc 1
LUZ_INT2	Permite conocer si una llamada ha sido realizada desde el interior del asc 2
PARA_ASC1	Relacionada con el boton S de la cabina 1
PARA_ASC2	Relacionada con el boton S de la cabina 2
REACTIV_ASC1	Relacionada con el boton A de la cabina 1
REACTIV_ASC2	Relacionada con el boton A de la cabina 2

Tabla 6.3 Variables diferentes o añadidas respecto a la simulación de la maniobra colectiva selectiva

Además, para esta maniobra de la maqueta se han realizado una tabla de animación para permitir observar a través del programa el estado de cada uno de los ascensores, conocer los registros de cada memoria, apreciar los leds encendidos o distinguir en qué planta se encuentra cada ascensor.

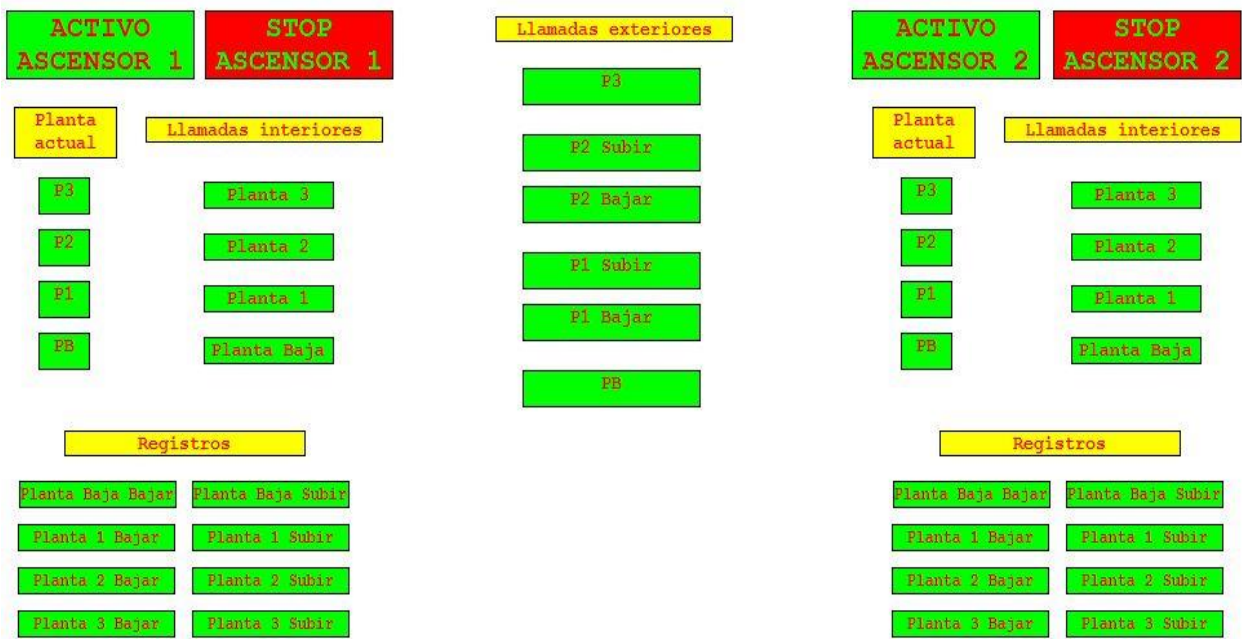


Figura 6.6 Tabla de animación utilizada en los ascensores con maniobra colectiva selectiva

7 CONCLUSIONES Y LÍNEAS DE MEJORA

7.1 Líneas de mejoras

Es evidente que los ascensores, como toda la tecnología utilizada, mejora a pasos agigantados con el transcurso de los años. En la actualidad, se construyen cada vez ascensores más modernos, con ahorros energéticos, menos contaminación, más silenciosos y un sinfín de particularidades que día a día permiten superar más barreras. Si esto lo aplicamos a este trabajo, la primera línea de mejora podría ser la creación de una nueva maqueta de ascensores, con más plantas, elementos más novedosos y menos ruidosos, una creación de puertas o incluso utilizar un sistema diferente de ascensor, como puede ser el sistema hidráulico. Por lo que, en lo referido a la maqueta, las líneas de mejoras tienden a infinito ya que cada usuario puede crear tantas maquetas diferentes como desee.

En lo que se refiere a la maqueta y el autómatas de los que se disponen para este proyecto, las principales líneas de mejora que se pueden realizar son la introducción de nuevas entradas que permitan más posibilidades a la hora de la recreación, como puede ser la inclusión de un mayor número de sensores para tener más precisión de la ubicación de cada cabina y así realizar detenciones menos bruscas. Otro aspecto que puede hacer elevar la funcionalidad de la maqueta es la instalación de una mayor cantidad de pulsadores, los cuales permitirían añadir nuevas funcionalidades a la maqueta. El hecho de incluir una mayor cantidad de entradas o salidas va ligado con la utilización de una cantidad mayor de módulos de las mismas ya que, incluso tal y como se puede apreciar en el proyecto actual, el número de salidas disponibles es demasiado reducido como para utilizar todas las opciones que la maqueta nos ofrece, por lo que la principal línea de mejora sería la inclusión de un nuevo módulo de salidas.

De la misma manera que se puede mejorar los elementos de los que se dispone y añadir otros nuevos, la programación del control del ascensor es otro tema que también permite una gran cantidad de variaciones respecto a lo que se tiene configurado y que no produciría un sobrecoste, a diferencia de las mejoras físicas de la maqueta y/o PLC propuestas. La implementación de una maniobra diferente es la principal mejora en este proyecto, ya que cada ascensor es diferente tanto en su estructura como en su control, por lo que la libertad para ser programados es inmensa.

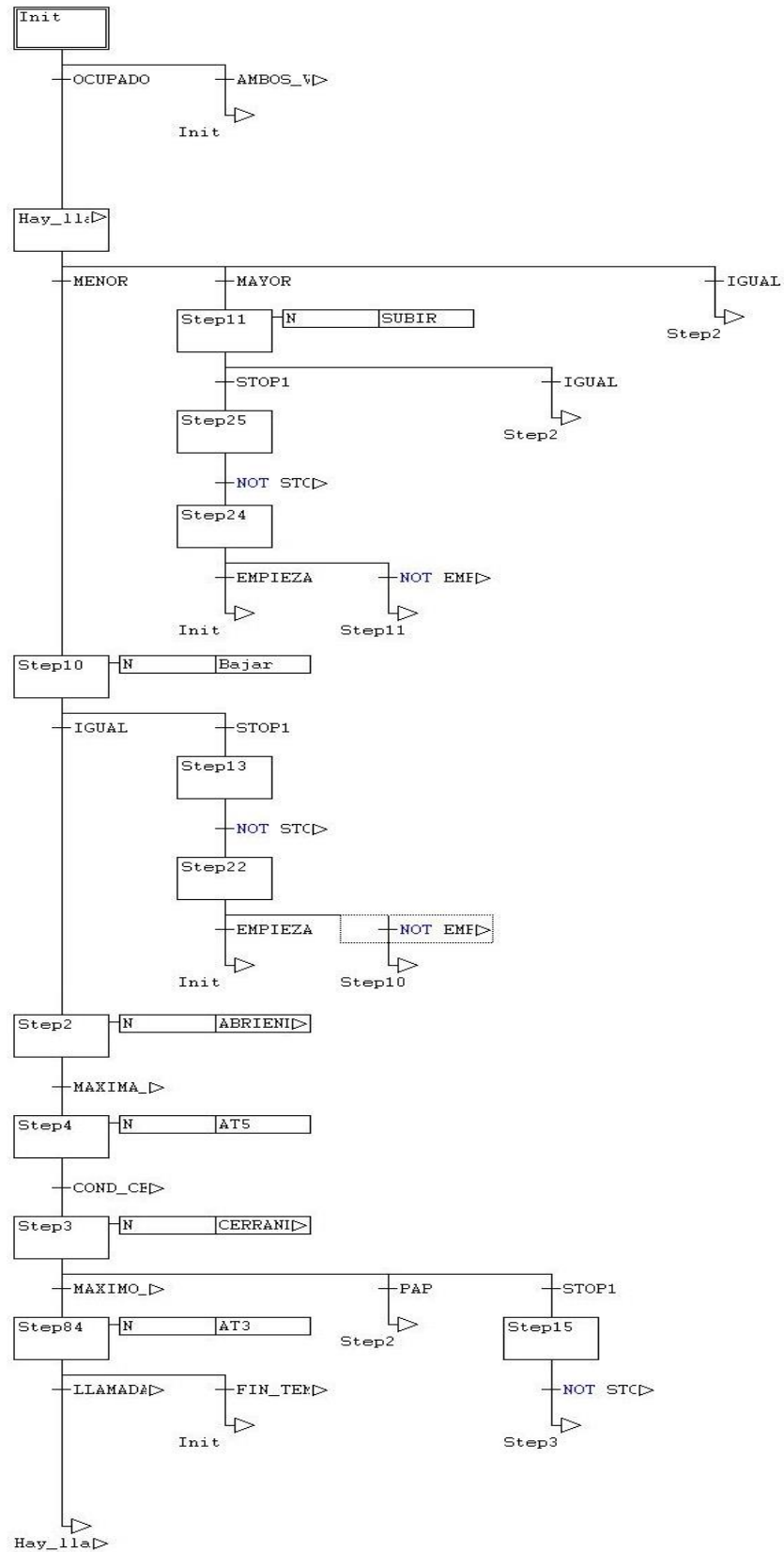
7.2 Conclusiones

La principal conclusión que se obtiene de este trabajo es que la automatización es implementable casi a cualquier elemento ya que se ha conseguido, realizando los cambios necesarios, poner en marcha de nuevo y automatizar un sistema de ascensores que, además de estar parado durante muchos años, no estaba preparado para ser utilizado con un autómatas programable.

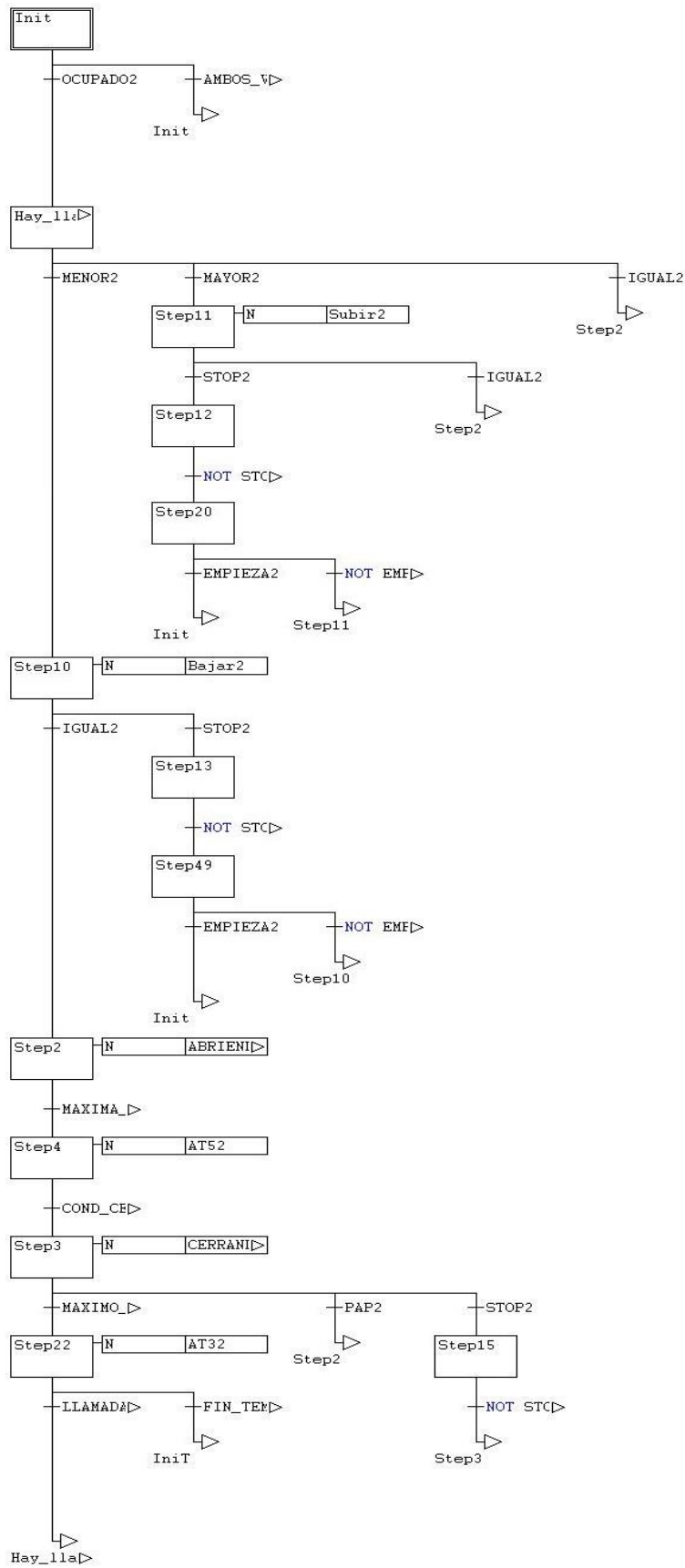
Además de esto, dentro del ámbito de la programación se observa que existe un amplio catálogo de posibilidades para lograr un mismo objetivo, que en este caso no es otro que el de controlar un ascensor. Como ya se ha dicho, cada ascensor es diferente por lo que existen multitud de variantes a la hora de ser controlados y que, una vez preparada la maqueta para ser puesta en marcha con un autómatas, ofrece la posibilidad de poder ser utilizada con otros fines educativos o incluso para la creación de nuevos proyectos.

ANEXO I: SIMULACIÓN MANIOBRA UNIVERSAL

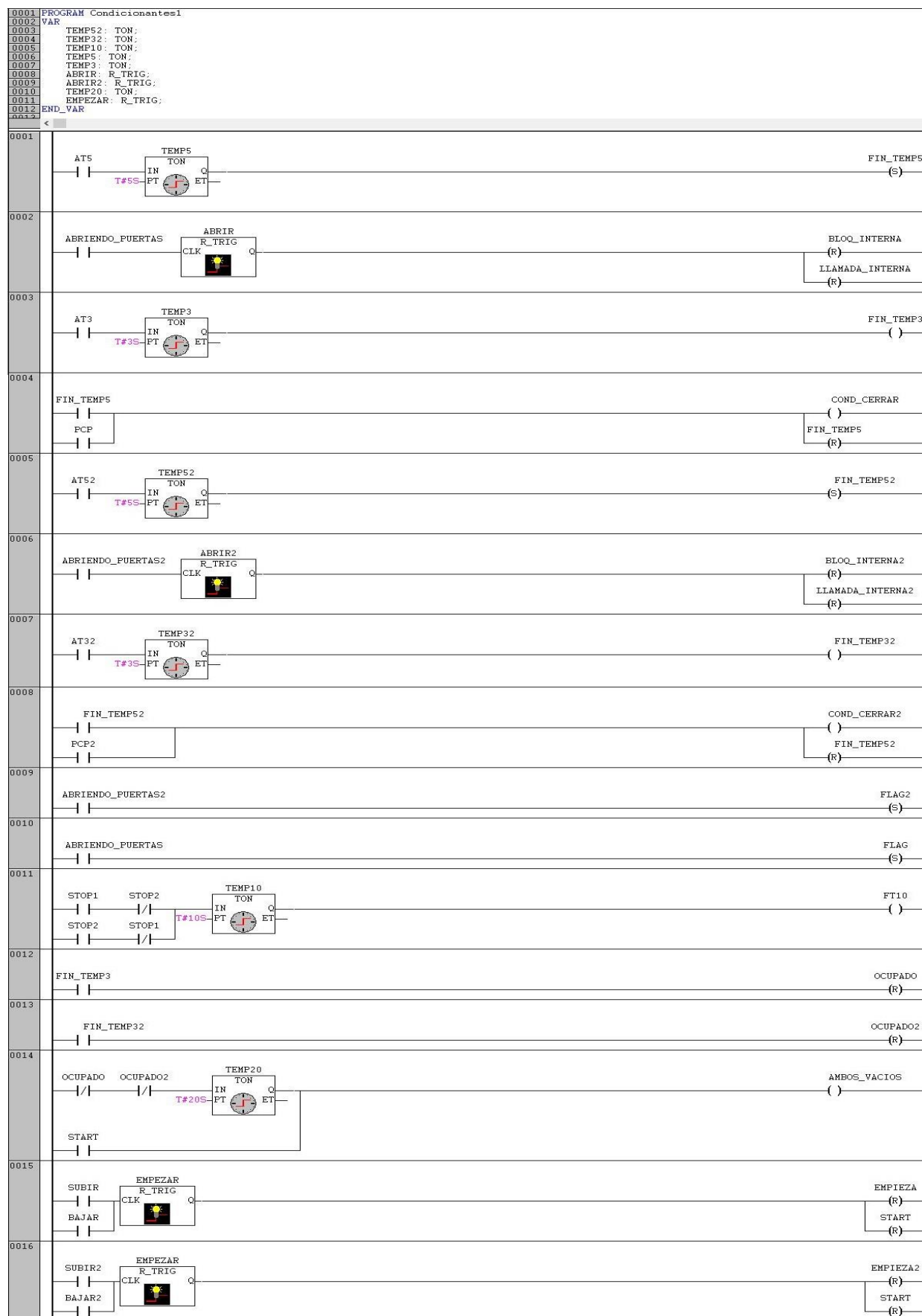
ASCENSOR 1 (SFC)



ASCENSOR 2 (SFC)



CONDICIONANTES1 (LD)



PARADAS (ST)

```

0001 PROGRAM PARADAS
0002 VAR
0003   i: INT;
0004   j: INT;
0005 END_VAR
0006
0007 <
0008
0001 (*Ascensor 1 y 2*)
0002
0003 (*Elección del ascensor que atiende la llamada si están los dos activos*)
0004 IF STOP1=FALSE AND STOP2=FALSE THEN
0005   FOR i:=1 TO 10 BY 1 DO
0006     IF PULSADOR_EXT[i]=1 AND LUZ[i]=FALSE THEN
0007       ALGO_PULSADO:=i;
0008       (*Si ambos ascensores están libres, atenderá la llamada el ascensor que se encuentre mas cerca del piso de destino*)
0009       IF OCUPADO=FALSE AND OCUPADO2=FALSE THEN
0010         LUZ[i]:=TRUE;
0011         IF ABS(PLANTA2-ALGO_PULSADO) < ABS(PLANTA-ALGO_PULSADO) THEN
0012           PISO_PULSADO2:=i;
0013           OCUPADO2:=TRUE;
0014           BLOQ_INTERNA2:=TRUE;
0015         ELSE
0016           PISO_PULSADO:=i;
0017           OCUPADO:=TRUE;
0018           BLOQ_INTERNA:=TRUE;
0019         END_IF
0020       (*Si un ascensor ya está ocupado, atenderá el otro la llamada*)
0021       ELSIF OCUPADO=FALSE AND OCUPADO2=TRUE THEN
0022         PISO_PULSADO:=i;
0023         OCUPADO:=TRUE;
0024         BLOQ_INTERNA:=TRUE;
0025       ELSIF OCUPADO2=FALSE AND OCUPADO=TRUE THEN
0026         PISO_PULSADO2:=i;
0027         OCUPADO2:=TRUE;
0028         BLOQ_INTERNA2:=TRUE;
0029       END_IF
0030     END_IF
0031   END_FOR
0032 END_IF
0033
0034
0035 (*Si ningún ascensor se encuentra disponible para atender una llamada,
0036 se encenderán todas las luces de los pasillos para hacerlo saber a los usuarios*)
0037 IF (STOP1=TRUE AND OCUPADO2=TRUE) OR (STOP2=TRUE AND OCUPADO=TRUE) OR
0038 (STOP1=TRUE AND STOP2=TRUE) OR (OCUPADO=TRUE AND OCUPADO2=TRUE) THEN
0039   FOR j:=1 TO 10 BY 1 DO
0040     LUZ[j]:=1;
0041   END_FOR
0042 END_IF
0043
0044
0045 (*Se apagarán las luces del interior de los ascensores en cuanto comience a abrirse sus puertas*)
0046 FOR i:=1 TO 10 BY 1 DO
0047   IF ABRIENDO_PUERTAS=TRUE AND SENSOR[i]=TRUE AND BLOQ_INTERNA=FALSE THEN
0048     LUZ_INT[i]:=FALSE;
0049   ELSIF ABRIENDO_PUERTAS2=TRUE AND SENSOR2[i]=TRUE AND BLOQ_INTERNA2=FALSE THEN
0050     LUZ_INT2[i]:=FALSE;
0051   END_IF
0052 END_FOR
0053
0054
0055 (*Apagado de luces de pasillo*)
0056 IF STOP1=FALSE AND STOP2=FALSE THEN
0057   (*Si ambos ascensores están libres, se apagan todas*)
0058   IF OCUPADO=FALSE AND OCUPADO2=FALSE THEN
0059     FOR i:=1 TO 10 BY 1 DO
0060       LUZ[i]:=FALSE;
0061     END_FOR
0062   (*Si uno está libre y el otro no, se mantendrá encendida la luz del pasillo
0063   que atiende el ascensor ocupado siempre que haya sido desde el exterior*)
0064   ELSIF OCUPADO=TRUE AND OCUPADO2=FALSE THEN
0065     FOR i:=1 TO 10 BY 1 DO
0066       IF LLAMADA_INTERNA=FALSE AND i<>PISO_PULSADO THEN
0067         LUZ[i]:=FALSE;
0068       ELSIF LLAMADA_INTERNA=TRUE THEN
0069         LUZ[i]:=FALSE;
0070       END_IF
0071     END_FOR
0072   ELSIF OCUPADO=FALSE AND OCUPADO2=TRUE THEN
0073     FOR i:=1 TO 10 BY 1 DO
0074       IF LLAMADA_INTERNA2=FALSE AND i<>PISO_PULSADO2 THEN
0075         LUZ[i]:=FALSE;
0076       ELSIF LLAMADA_INTERNA2=TRUE THEN
0077         LUZ[i]:=FALSE;
0078       END_IF
0079     END_FOR
0080   END_IF
0081 (*Si uno de los ascensores está vacío y el otro con parada forzada*)
0082 ELSIF STOP1=FALSE AND STOP2=TRUE THEN
0083   IF OCUPADO=FALSE THEN
0084     FOR i:=1 TO 10 BY 1 DO
0085       LUZ[i]:=FALSE;
0086     END_FOR
0087   END_IF
0088 ELSIF STOP2=FALSE AND STOP1=TRUE THEN
0089   IF OCUPADO2=FALSE THEN
0090     FOR i:=1 TO 10 BY 1 DO
0091       LUZ[i]:=FALSE;
0092     END_FOR
0093   END_IF
0094 END_IF
0095
0096
0097 (*ASCENSOR 1*)
0098
0099 (*Si se produce una llamada interna mientras sea posible su realización, enciende la luz de la cabina de la planta
0100 de destino e impide que el ascensor pueda ser llamado de nuevo tanto desde el interior como desde el exterior*)
0101 FOR i:=1 TO 10 BY 1 DO
0102   IF PULSADOR_INT[i]=TRUE AND FLAG=TRUE AND BLOQ_INTERNA=FALSE THEN
0103     LLAMADA_INTERNA:=TRUE;
0104     BLOQ_INTERNA:=TRUE;
0105     PISO_PULSADO:=i;
0106     OCUPADO:=TRUE;

```

```

0107     LUZ_INT[i]:=TRUE;
0108     END_IF
0109 END_FOR
0110
0111
0112 (*Actualiza la planta actual cada vez que llega al sensor de una planta*)
0113 FOR i:=1 TO 10 BY 1 DO
0114     IF SENSOR[i]=TRUE THEN
0115         PLANTA:=i;
0116     END_IF
0117 END_FOR
0118
0119
0120
0121 (*El ascensor 1 atenderá todas las llamadas mientras el ascensor 2 esta parado de manera forzada*)
0122 IF STOP2=TRUE THEN
0123     FOR i:=1 TO 10 BY 1 DO
0124         IF PULSADOR_EXT[i]=TRUE AND OCUPADO=FALSE THEN
0125             PISO_PULSADO:=i;
0126             OCUPADO:=TRUE;
0127             BLOQ_INTERNA:=TRUE;
0128             ELSIF PULSADOR_INT1[i]=TRUE AND BLOQ_INTERNA=FALSE AND FLAG=TRUE THEN
0129                 PISO_PULSADO:=i;
0130                 OCUPADO:=TRUE;
0131                 BLOQ_INTERNA:=TRUE;
0132                 LLAMADA_INTERNA:=TRUE;
0133             END_IF
0134         END_FOR
0135     END_IF
0136
0137
0138 (*Si el ascensor 2 está parado forzamente y realizaba una llamada de pasillo, si se han cumplido los 10 segundos y
0139 el ascensor 1 está libre y en funcionamiento, recibirá el traspaso de la llamada del ascensor 2 para que no se pierda*)
0140 IF STOP2=TRUE AND STOP1=FALSE AND OCUPADO2=TRUE AND OCUPADO=FALSE AND FT10=TRUE AND LLAMADA_INTERNA2=FALSE THEN
0141     PISO_PULSADO:=PISO_PULSADO2;
0142     OCUPADO2:=FALSE;
0143     OCUPADO:=TRUE;
0144     BLOQ_INTERNA:=TRUE;
0145     BLOQ_INTERNA2:=FALSE;
0146     EMPIEZA2:=TRUE;
0147 END_IF
0148
0149
0150 (*Elección del movimiento en función de la planta actual y el piso de destino*)
0151 IF PISO_PULSADO>PLANTA THEN
0152     MAYOR:=1;
0153     MENOR:=0;
0154     IGUAL:=0;
0155 ELSIF PLANTA>PISO_PULSADO AND PISO_PULSADO<>0 THEN
0156     MENOR:=1;
0157     MAYOR:=0;
0158     IGUAL:=0;
0159 ELSE
0160     IGUAL:=1;
0161     MAYOR:=0;
0162     MENOR:=0;
0163 END_IF
0164
0165
0166 (*Si ambos ascensores están libres durante 40 segundos y el ascensor 1 no se encuentra en la planta 1,
0167 tomará dicha planta como destino simulando una llamada*)
0168 IF AMBOS_VACIOS=TRUE AND SENSOR[1]=FALSE THEN
0169     LLAMADA_INTERNA:=TRUE;
0170     BLOQ_INTERNA:=TRUE;
0171     PISO_PULSADO:=1;
0172     OCUPADO:=TRUE;
0173     LUZ_INT[1]:=TRUE;
0174 END_IF
0175
0176
0177
0178 (*ASCENSOR 2*)
0179 (*Mismos casos y ordenados igual que el ascensor 1*)
0180
0181 FOR i:=1 TO 10 BY 1 DO
0182     IF PULSADOR_INT2[i]=TRUE AND BLOQ_INTERNA2=FALSE AND FLAG2=TRUE THEN
0183         LLAMADA_INTERNA2:=TRUE;
0184         BLOQ_INTERNA2:=TRUE;
0185         PISO_PULSADO2:=i;
0186         OCUPADO2:=TRUE;
0187         LUZ_INT2[i]:=TRUE;
0188     END_IF
0189 END_FOR
0190
0191
0192 FOR i:=1 TO 10 BY 1 DO
0193     IF SENSOR2[i]=1 THEN
0194         PLANTA2:=i;
0195     END_IF
0196 END_FOR
0197
0198
0199 IF STOP1=TRUE THEN
0200     FOR i:=1 TO 10 BY 1 DO
0201         IF PULSADOR_EXT[i]=TRUE AND OCUPADO2=FALSE THEN
0202             PISO_PULSADO2:=i;
0203             OCUPADO2:=TRUE;
0204             BLOQ_INTERNA2:=TRUE;
0205             ELSIF PULSADOR_INT2[i]=TRUE AND BLOQ_INTERNA2=FALSE AND FLAG2=TRUE THEN
0206                 PISO_PULSADO2:=i;
0207                 OCUPADO2:=TRUE;
0208                 BLOQ_INTERNA2:=TRUE;
0209                 LLAMADA_INTERNA2:=TRUE;
0210             END_IF
0211         END_FOR
0212     END_IF
0213
0214
0215 IF STOP1=TRUE AND STOP2=FALSE AND OCUPADO=TRUE AND OCUPADO2=FALSE AND FT10=TRUE AND LLAMADA_INTERNA=FALSE THEN
0216     PISO_PULSADO2:=PISO_PULSADO;
0217     OCUPADO:=FALSE;
0218     OCUPADO2:=TRUE;
0219     BLOQ_INTERNA2:=TRUE;
0220     BLOQ_INTERNA:=FALSE;
0221     EMPIEZA:=TRUE;
0222 END_IF
0223
0224
0225 IF PISO_PULSADO2>PLANTA2 THEN
0226     MAYOR2:=1;
0227     MENOR2:=0;
0228     IGUAL2:=0;
0229 ELSIF PLANTA2>PISO_PULSADO2 AND PISO_PULSADO2<>0 THEN
0230     MENOR2:=1;
0231     MAYOR2:=0;
0232     IGUAL2:=0;
0233 ELSE
0234     IGUAL2:=1;
0235     MAYOR2:=0;
0236     MENOR2:=0;
0237 END_IF
0238
0239
0240 IF AMBOS_VACIOS=TRUE AND SENSOR2[5]=FALSE THEN
0241     LLAMADA_INTERNA2:=TRUE;
0242     BLOQ_INTERNA2:=TRUE;
0243     PISO_PULSADO2:=5;
0244     OCUPADO2:=TRUE;
0245     LUZ_INT2[5]:=TRUE;
0246 END_IF

```


SIMULACIÓN (ST)

```

0001 PROGRAM simulacion
0002 VAR
0003     ASCENSOR_1: LREAL:=0;
0004     incremento: INT:=1;
0005     i: INT;
0006     j: INT;
0007     x: INT:=1;
0008     ASCENSOR_2: LREAL:=0;
0009     n: INT:=1;
0010 END_VAR
0011
0012 <
0013
0001 (*ASCENSOR 1*)
0002
0003 IF ABRIENDO_PUERTAS=1 THEN
0004     FOR i:=1 TO 10 BY 1 DO
0005         IF SENSOR[i]=1 AND MAX_APE[i]=FALSE THEN
0006             PUERTA_DER[i]:=PUERTA_DER[i]+incremento;
0007             PUERTA_IZQ[i]:=PUERTA_IZQ[i]-incremento;
0008         END_IF
0009     END_FOR
0010 END_IF
0011
0012 IF CERRANDO_PUERTAS=1 THEN
0013     FOR i:=1 TO 10 BY 1 DO
0014         IF SENSOR[i]=1 AND MAX_CIE[i]=FALSE THEN
0015             PUERTA_DER[i]:=PUERTA_DER[i]-incremento;
0016             PUERTA_IZQ[i]:=PUERTA_IZQ[i]+incremento;
0017         END_IF
0018     END_FOR
0019 END_IF
0020
0021
0022 FOR i:=1 TO 10 BY 1 DO
0023     IF PUERTA_DER[i]<=0 AND PUERTA_IZQ[i]>=0 THEN
0024         MAX_CIE[i]:=TRUE;
0025     ELSEIF PUERTA_DER[i]>=70 AND PUERTA_IZQ[i]<=-70 THEN
0026         MAX_APE[i]:=TRUE;
0027     ELSE
0028         MAX_APE[i]:=FALSE;
0029         MAX_CIE[i]:=FALSE;
0030         x:=i;
0031     END_IF
0032 END_FOR
0033
0034
0035 IF MAX_APE[x]=FALSE THEN
0036     MAXIMA_APERTURA:=FALSE;
0037 ELSE
0038     MAXIMA_APERTURA:=TRUE;
0039 END_IF
0040
0041
0042 IF MAX_CIE[x]=FALSE THEN
0043     MAXIMO_CIERRE:=FALSE;
0044 ELSE
0045     MAXIMO_CIERRE:=TRUE;
0046 END_IF
0047
0048
0049 IF Subir=1 THEN
0050     ASCENSOR_1:=ASCENSOR_1-0.75;
0051 END_IF
0052
0053 IF Bajar=1 THEN
0054     ASCENSOR_1:=ASCENSOR_1+0.75;
0055 END_IF
0056
0057
0058 FOR i:=1 TO 10 BY 1 DO
0059     IF ASCENSOR_1<=1.5-(i-1)*120 AND ASCENSOR_1>=-1.5-(i-1)*120 THEN
0060         SENSOR[i]:=1;
0061         PLANTA:=i;
0062     ELSE
0063         SENSOR[i]:=0;
0064     END_IF
0065 END_FOR
0066
0067
0068 FOR i:=1 TO 10 BY 1 DO
0069     IF (Subir=1 AND ASCENSOR_1<=5-(i-1)*120 AND ASCENSOR_1>=-115-(i-1)*120) OR
0070        (ASCENSOR_1>=-2-(i-1)*120 AND ASCENSOR_1<=2-(i-1)*120) OR
0071        (Bajar=1 AND ASCENSOR_1>=-4-(i-1)*120 AND ASCENSOR_1<=-4-(i-2)*120) THEN
0072         DISPLAY1:=i;
0073     END_IF
0074 END_FOR
0075
0076
0077
0078 (*ASCENSOR 2*)
0079
0080 IF ABRIENDO_PUERTAS2=1 THEN
0081     FOR i:=1 TO 10 BY 1 DO
0082         IF SENSOR2[i]=1 AND MAX_APE2[i]=FALSE THEN
0083             PUERTA_DER2[i]:=PUERTA_DER2[i]+incremento;
0084             PUERTA_IZQ2[i]:=PUERTA_IZQ2[i]-incremento;
0085         END_IF
0086     END_FOR
0087 END_IF
0088
0089

```



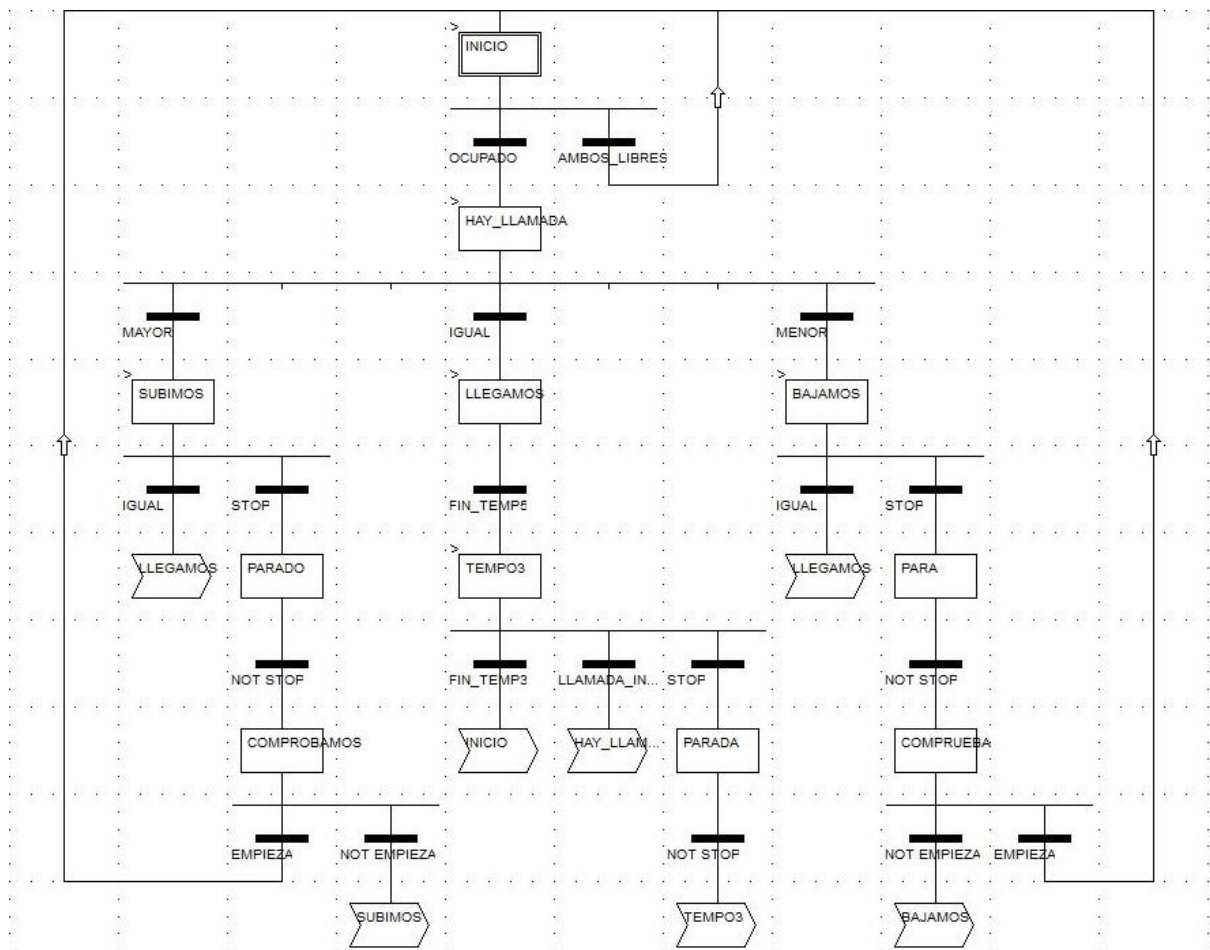
```

0090
0091 IF CERRANDO_PUERTAS2=1 THEN
0092     FOR i:=1 TO 10 BY 1 DO
0093         IF SENSOR2[i]=1 AND MAX_CIE2[i]=FALSE THEN
0094             PUERTA_DER2[i]:=PUERTA_DER2[i]-incremento;
0095             PUERTA_IZQ2[i]:=PUERTA_IZQ2[i]+incremento;
0096         END_IF
0097     END_FOR
0098 END_IF
0099
0100
0101 FOR i:=1 TO 10 BY 1 DO
0102     IF PUERTA_DER2[i]<=0 AND PUERTA_IZQ2[i]>=0 THEN
0103         MAX_CIE2[i]:=TRUE;
0104     ELSEIF PUERTA_DER2[i]>=70 AND PUERTA_IZQ2[i]<=-70 THEN
0105         MAX_APE2[i]:=TRUE;
0106     ELSE
0107         MAX_APE2[i]:=FALSE;
0108         MAX_CIE2[i]:=FALSE;
0109         n:=i;
0110     END_IF
0111 END_FOR
0112
0113
0114 IF MAX_APE2[n]=FALSE THEN
0115     MAXIMA_APERTURA2:=FALSE;
0116 ELSE
0117     MAXIMA_APERTURA2:=TRUE;
0118 END_IF
0119
0120
0121 IF MAX_CIE2[n]=FALSE THEN
0122     MAXIMO_CIERRE2:=FALSE;
0123 ELSE
0124     MAXIMO_CIERRE2:=TRUE;
0125 END_IF
0126
0127
0128 IF Subir2=1 THEN
0129     ASCENSOR_2:=ASCENSOR_2-0.75;
0130 END_IF
0131
0132 IF Bajar2=1 THEN
0133     ASCENSOR_2:=ASCENSOR_2+0.75;
0134 END_IF
0135
0136
0137 FOR i:=1 TO 10 BY 1 DO
0138     IF ASCENSOR_2<=1.5-(i-1)*120 AND ASCENSOR_2>=-1.5-(i-1)*120 THEN
0139         SENSOR2[i]:=1;
0140         PLANTA2:=i;
0141     ELSE
0142         SENSOR2[i]:=0;
0143     END_IF
0144 END_FOR
0145
0146
0147 FOR i:=1 TO 10 BY 1 DO
0148     IF (Subir2=1 AND ASCENSOR_2<=5-(i-1)*120 AND ASCENSOR_2>=-115-(i-1)*120) OR
0149        (ASCENSOR_2>=-2-(i-1)*120 AND ASCENSOR_2<=2-(i-1)*120) OR
0150        (Bajar2=1 AND ASCENSOR_2>=-4-(i-1)*120 AND ASCENSOR_2<=-4-(i-2)*120) THEN
0151         DISPLAY2:=i;
0152     END_IF
0153 END_FOR
0154

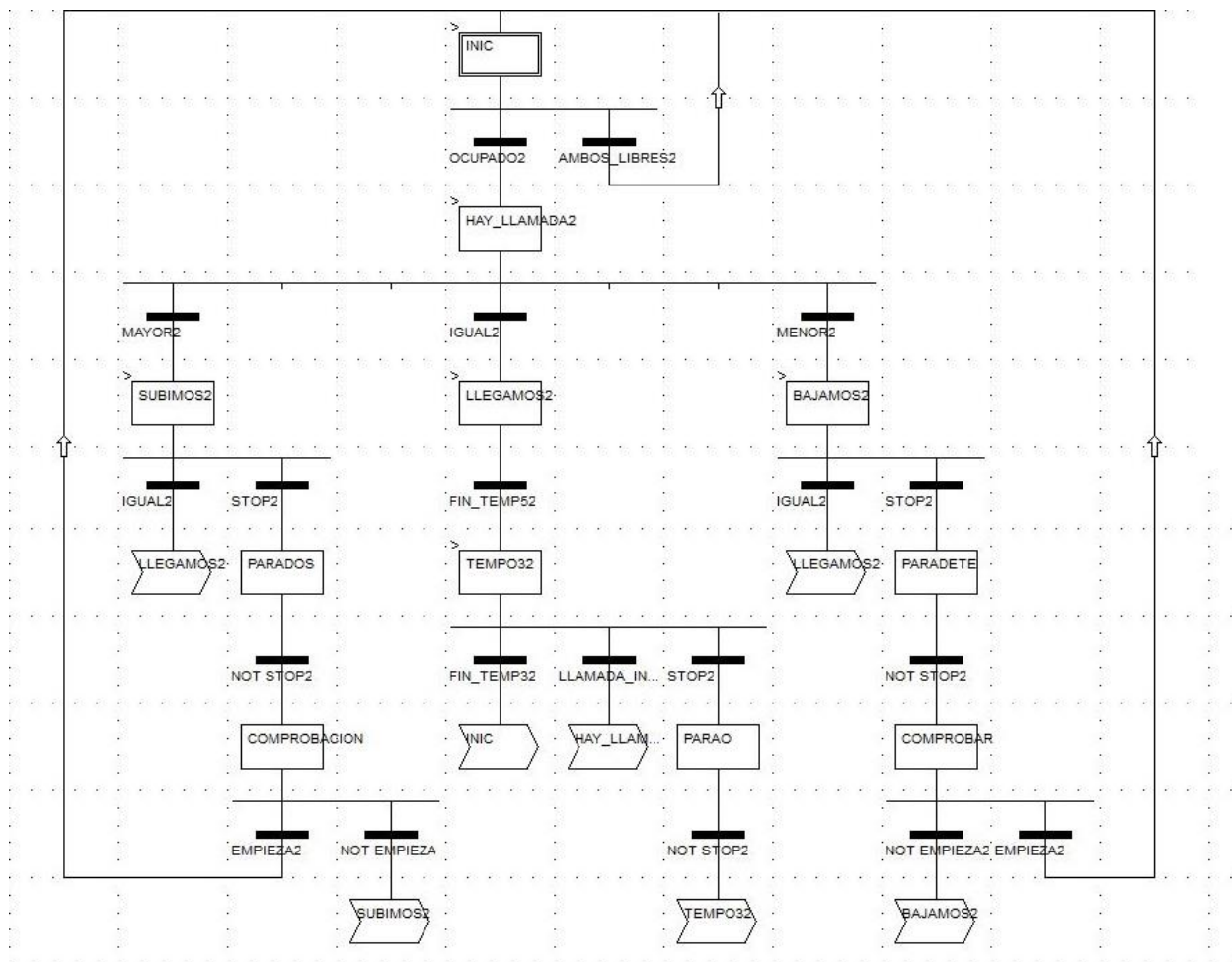
```

ANEXO II: MAQUETA MANIOBRA UNIVERSAL

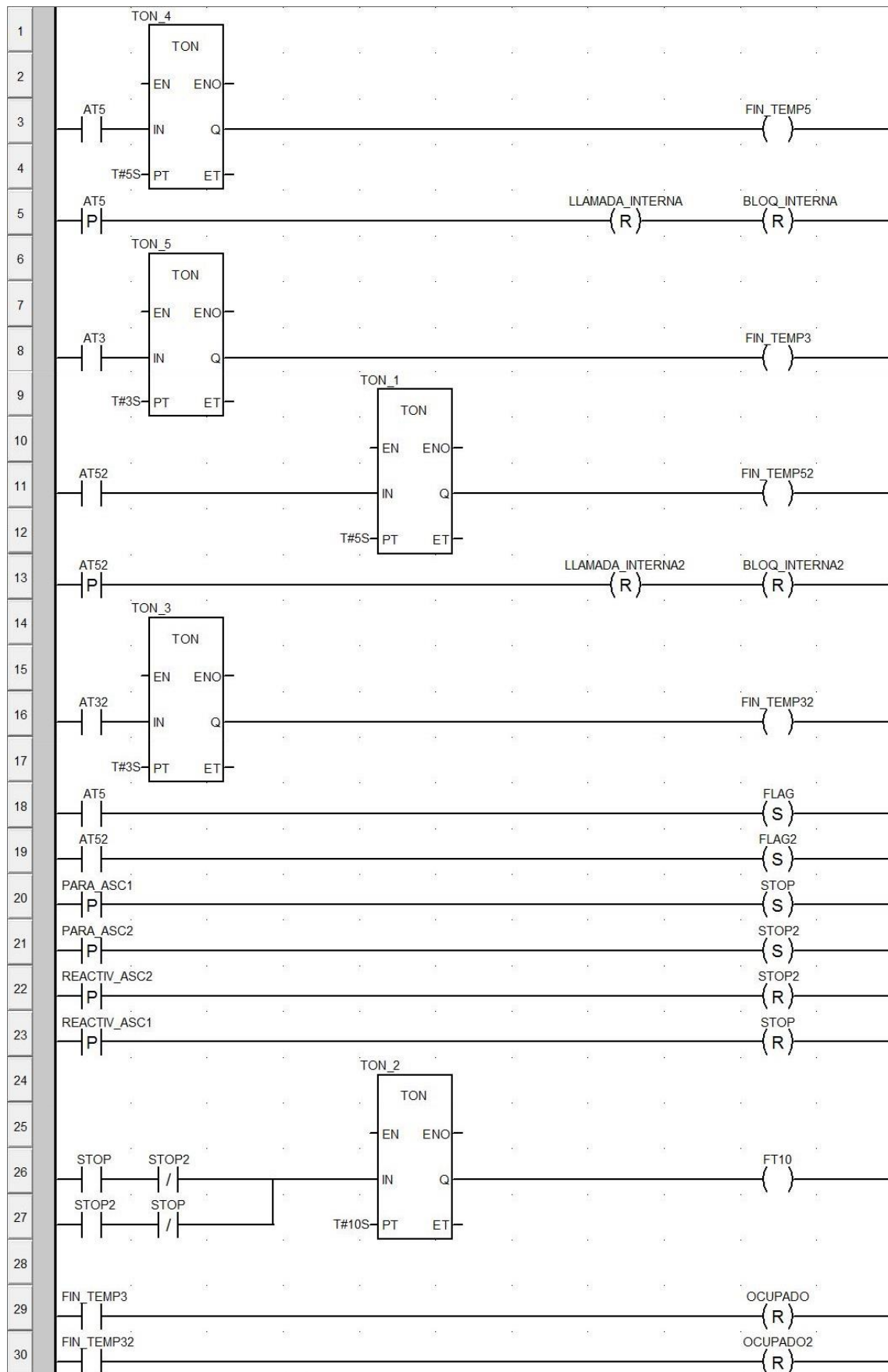
ASCENSOR 1 (SFC)

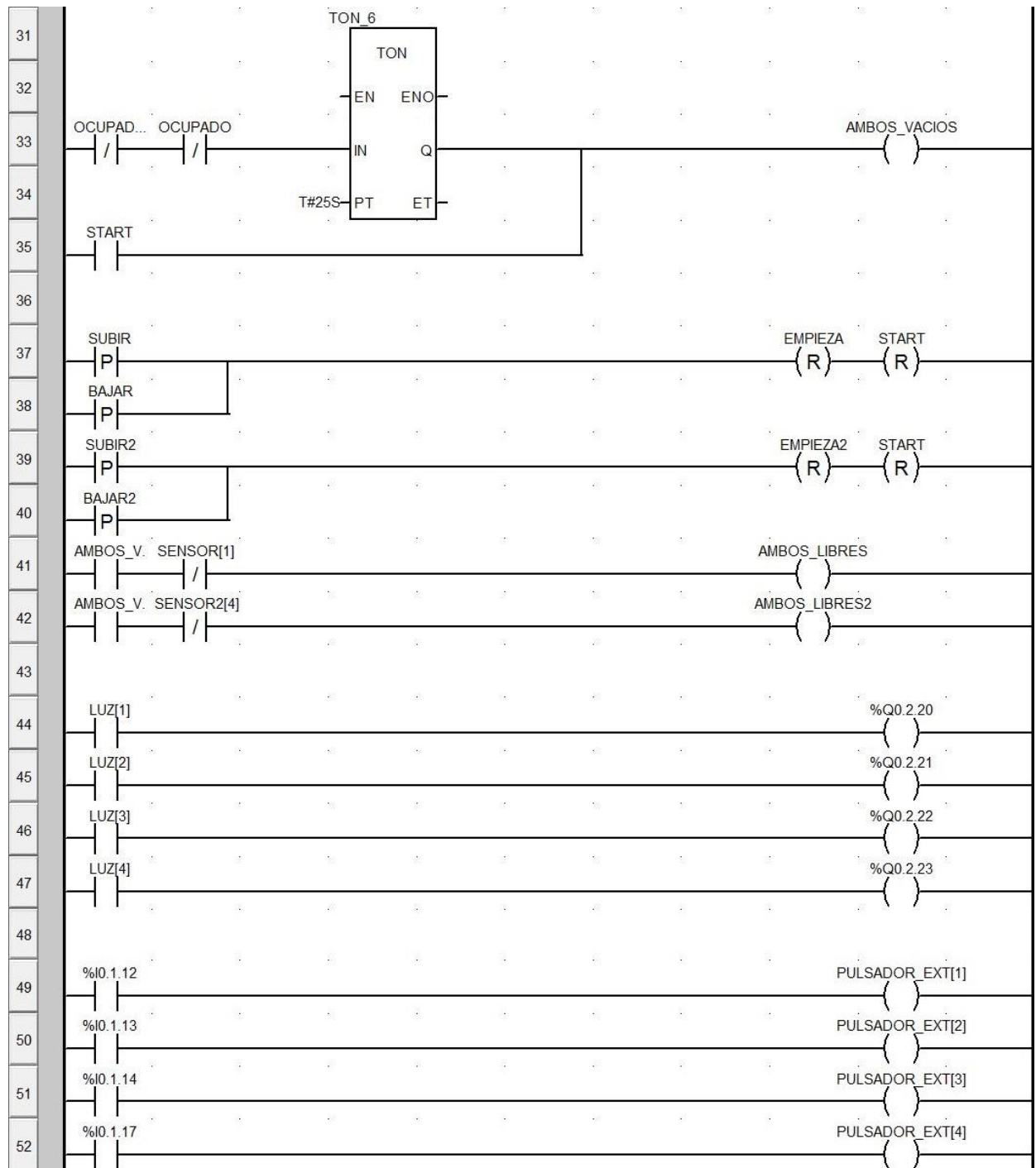


ASCENSOR 2 (SFC)



CONDICIONES (LD)





PARADAS (ST)

```
(*Ascensor 1 y 2*)

(*Elección del ascensor que atiende la llamada si están los dos activos*)
IF STOP=FALSE AND STOP2=FALSE THEN
  FOR i:=1 TO 4 BY 1 DO
    IF PULSADOR_EXT[i]=1 AND LUZ[i]=FALSE THEN
      ALGO_PULSADO:=i;
      (*Si ambos ascensores están libres, atenderá la llamada el ascensor mas cercano al piso de destino*)
      IF OCUPADO=FALSE AND OCUPADO2=FALSE THEN
        LUZ[i]:=TRUE;
        IF ABS(PLANTA2-ALGO_PULSADO) < ABS(PLANTA-ALGO_PULSADO) THEN
          PISO_PULSADO:=i;
          OCUPADO:=TRUE;
          BLOQ_INTERNA:=TRUE;
        ELSE
          PISO_PULSADO:=i;
          OCUPADO:=TRUE;
          BLOQ_INTERNA:=TRUE;
        END IF;
      (*Si un ascensor ya está ocupado, atenderá el otro la llamada*)
      ELIF OCUPADO=FALSE AND OCUPADO2=TRUE THEN
        PISO_PULSADO:=i;
        OCUPADO:=TRUE;
        BLOQ_INTERNA:=TRUE;
      ELIF OCUPADO2=FALSE AND OCUPADO=TRUE THEN
        PISO_PULSADO:=i;
        OCUPADO2:=TRUE;
        BLOQ_INTERNA2:=TRUE;
      END IF;
    END IF;
  END FOR;
END_IF;

(*Apagado de luces de pasillo*)
IF STOP=FALSE AND STOP2=FALSE THEN
  (*Si ambos ascensores están libres, se apagan todas*)
  IF OCUPADO=FALSE AND OCUPADO2=FALSE THEN
    FOR i:=1 TO 10 BY 1 DO
      LUZ[i]:=FALSE;
    END FOR;
  (*Si uno está libre y el otro no, se mantendrá encendida la luz del pasillo que atiende el ascensor ocupado siempre haya sido desde el exterior*)
  ELIF OCUPADO=TRUE AND OCUPADO2=FALSE THEN
    FOR i:=1 TO 4 BY 1 DO
      IF LLAMADA_INTERNA=FALSE AND i<>PISO_PULSADO THEN
        LUZ[i]:=FALSE;
      ELIF LLAMADA_INTERNA=TRUE THEN
        LUZ[i]:=FALSE;
      END IF;
    END FOR;
  ELIF OCUPADO=FALSE AND OCUPADO2=TRUE THEN
    FOR i:=1 TO 4 BY 1 DO
      IF LLAMADA_INTERNA2=FALSE AND i<>PISO_PULSADO2 THEN
        LUZ[i]:=FALSE;
      ELIF LLAMADA_INTERNA2=TRUE THEN
        LUZ[i]:=FALSE;
      END IF;
    END FOR;
  END IF;
END_IF;

(*Si uno de los ascensores está vacío y el otro con parada forzada*)
ELIF STOP=FALSE AND STOP2=TRUE THEN
  IF OCUPADO=FALSE THEN
    FOR i:=1 TO 4 BY 1 DO
      LUZ[i]:=FALSE;
    END FOR;
  END IF;
  ELIF STOP2=FALSE AND STOP=TRUE THEN
    IF OCUPADO2=FALSE THEN
      FOR i:=1 TO 4 BY 1 DO
        LUZ[i]:=FALSE;
      END FOR;
    END IF;
  END_IF;
END_IF;

(*ASCENSOR 1*)

(*Si se produce una llamada interna mientras sea posible su realización, enciende la luz de la cabina de la planta de destino e impide que el ascensor pueda ser llamado de nuevo tanto desde el interior como desde el exterior*)
FOR i:=1 TO 4 BY 1 DO
  IF PULSADOR_INT1[i]=TRUE AND FLAG=TRUE AND BLOQ_INTERNA=FALSE THEN
    LLAMADA_INTERNA:=TRUE;
    BLOQ_INTERNA:=TRUE;
    PISO_PULSADO:=i;
    OCUPADO:=TRUE;
  END IF;
END FOR;

(*Reconocimiento de planta*)
FOR i:=1 TO 4 BY 1 DO
  IF SENSOR[i]=TRUE THEN
    PLANTA:=i;
  END IF;
END FOR;

(*El ascensor 1 atenderá todas las llamadas mientras el ascensor 2 esta parado de manera forzada*)
IF STOP2=TRUE THEN
  FOR i:=1 TO 4 BY 1 DO
    IF PULSADOR_EXT[i]=TRUE AND OCUPADO=FALSE THEN
      PISO_PULSADO:=i;
      OCUPADO:=TRUE;
      BLOQ_INTERNA:=TRUE;
    ELIF PULSADOR_INT1[i]=TRUE AND BLOQ_INTERNA=FALSE AND FLAG=TRUE THEN
      PISO_PULSADO:=i;
      OCUPADO:=TRUE;
      BLOQ_INTERNA:=TRUE;
      LLAMADA_INTERNA:=TRUE;
    END IF;
  END FOR;
END_IF;
```



```

(*Si el ascensor 2 está parado forzamente y realizaba una llamada de pasillo,
si se han cumplido los 10 segundos y el ascensor 1 está libre y en funcionamiento,
recibirá el traspaso de la llamada del ascensor 2 para que no se pierda*)
IF STOP2=TRUE AND STOP=FALSE AND OCUPADO2=TRUE AND OCUPADO=FALSE AND FT10=TRUE AND LLAMADA_INTERNA2=FALSE THEN
    PISO_PULSADO:=PISO_PULSADO2;
    OCUPADO2:=FALSE;
    OCUPADO:=TRUE;
    BLOQ_INTERNA:=TRUE;
    BLOQ_INTERNA2:=FALSE;
    EMPIEZA2:=TRUE;
END_IF;

(*Elección del movimiento en función de la planta actual y el piso de destino*)
IF PISO_PULSADO>PLANTA THEN
    MAYOR:=1;
    MENOR:=0;
    IGUAL:=0;
ELSIF PLANTA>PISO_PULSADO AND PISO_PULSADO<>0 THEN
    MENOR:=1;
    MAYOR:=0;
    IGUAL:=0;
ELSE
    IGUAL:=1;
    MAYOR:=0;
    MENOR:=0;
END_IF;

(*Si ambos ascensores están libres durante 40 segundos y el ascensor 1 no se encuentra
en la planta 1, tomará dicha planta como destino simulando una llamada*)
IF AMBOS_VACIOS AND SENSOR[1]=FALSE THEN
    LLAMADA_INTERNA:=TRUE;
    BLOQ_INTERNA:=TRUE;
    PISO_PULSADO:=1;
    OCUPADO:=TRUE;
END_IF;

(*ASCENSOR 2*)
FOR i:=1 TO 4 BY 1 DO
    IF PULSADOR_INT2[i]=TRUE AND BLOQ_INTERNA2=FALSE AND FLAG2=TRUE THEN
        LLAMADA_INTERNA2:=TRUE;
        BLOQ_INTERNA2:=TRUE;
        PISO_PULSADO2:=i;
        OCUPADO2:=TRUE;
    END_IF;
END_FOR;

FOR i:=1 TO 4 BY 1 DO
    IF SENSOR2[i]=TRUE THEN
        PLANTA2:=i;
    END_IF;
END_FOR;

IF STOP=TRUE THEN
    FOR i:=1 TO 4 BY 1 DO
        IF PULSADOR_EXT[i]=TRUE AND OCUPADO2=FALSE THEN
            PISO_PULSADO2:=i;
            OCUPADO2:=TRUE;
            BLOQ_INTERNA2:=TRUE;
            LLAMADA_INTERNA2:=TRUE;
        END_IF;
    END_FOR;
END_IF;

IF STOP=TRUE AND STOP2=FALSE AND OCUPADO=TRUE AND OCUPADO2=FALSE AND FT10=TRUE AND LLAMADA_INTERNA=FALSE THEN
    PISO_PULSADO2:=PISO_PULSADO;
    OCUPADO:=FALSE;
    OCUPADO2:=TRUE;
    BLOQ_INTERNA2:=TRUE;
    BLOQ_INTERNA:=FALSE;
    EMPIEZA:=TRUE;
END_IF;

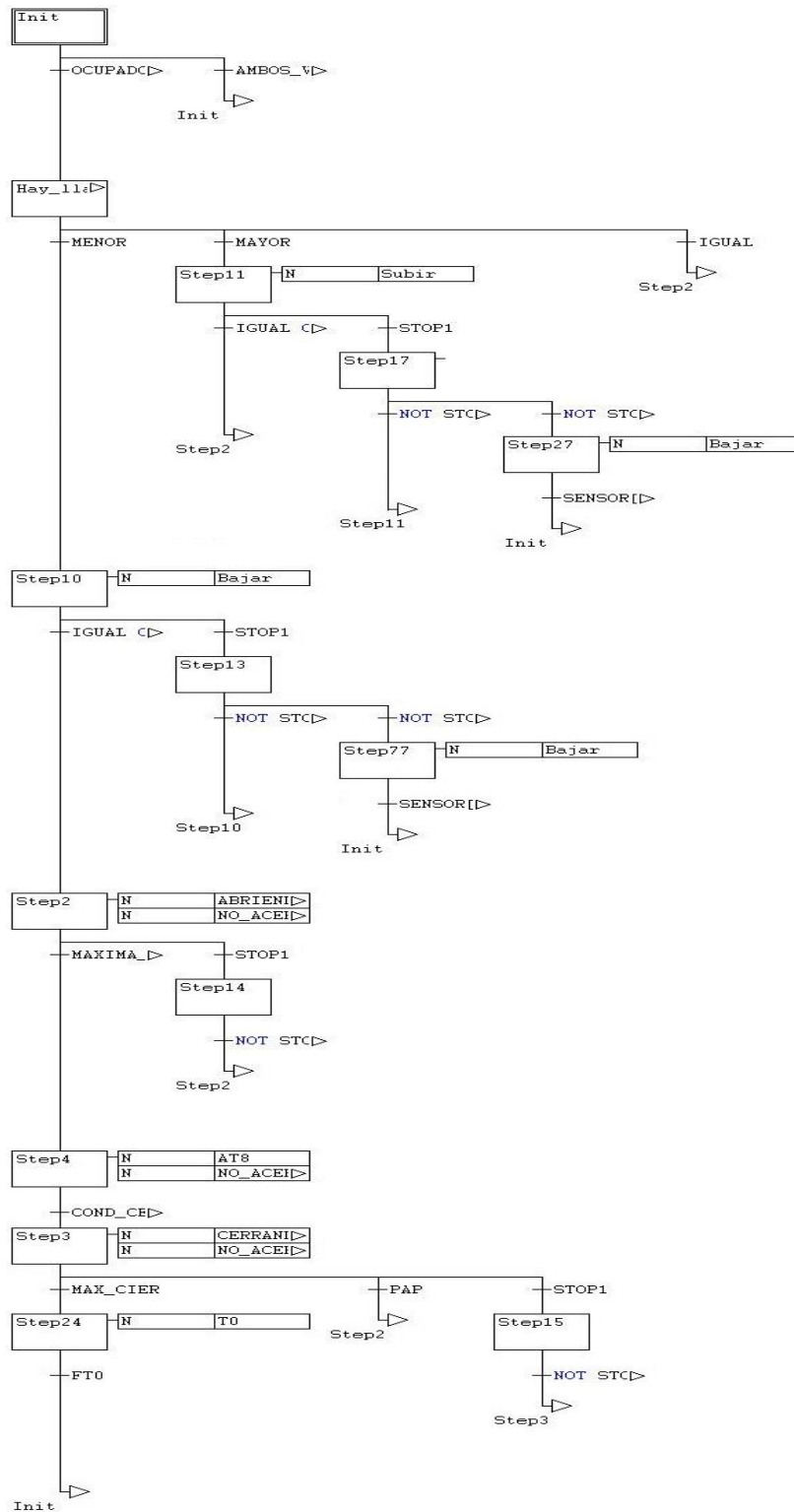
IF PISO_PULSADO2>PLANTA2 THEN
    MAYOR2:=1;
    MENOR2:=0;
    IGUAL2:=0;
ELSIF PLANTA2>PISO_PULSADO2 AND PISO_PULSADO2<>0 THEN
    MENOR2:=1;
    MAYOR2:=0;
    IGUAL2:=0;
ELSE
    IGUAL2:=1;
    MAYOR2:=0;
    MENOR2:=0;
END_IF;

IF AMBOS_VACIOS AND SENSOR2[4]=FALSE THEN
    LLAMADA_INTERNA2:=TRUE;
    BLOQ_INTERNA2:=TRUE;
    PISO_PULSADO2:=4;
    OCUPADO2:=TRUE;
END_IF;

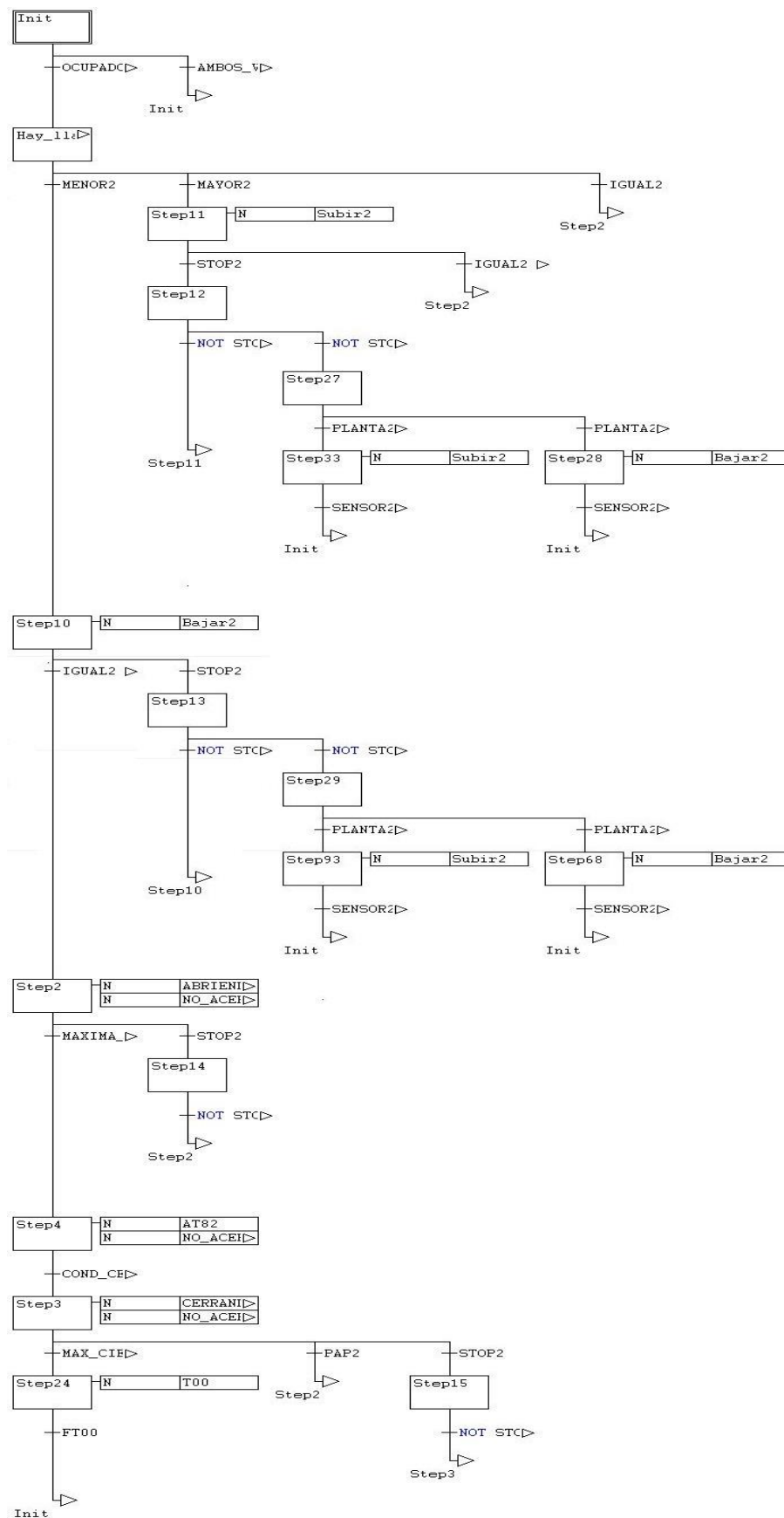
```

ANEXO III: SIMULACIÓN MANIOBRA COLECTIVA SELECTIVA

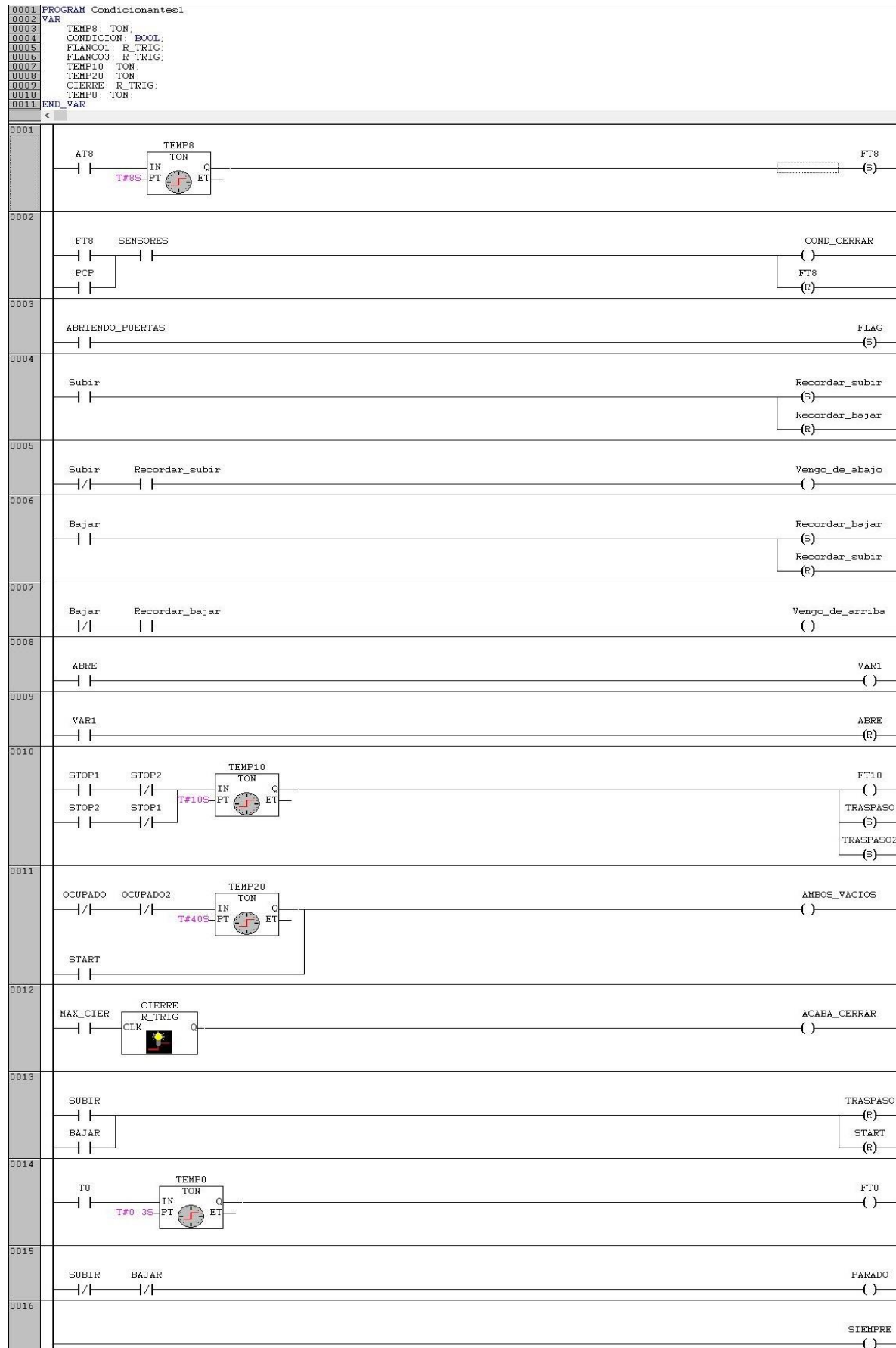
ELEVADOR (SFC)



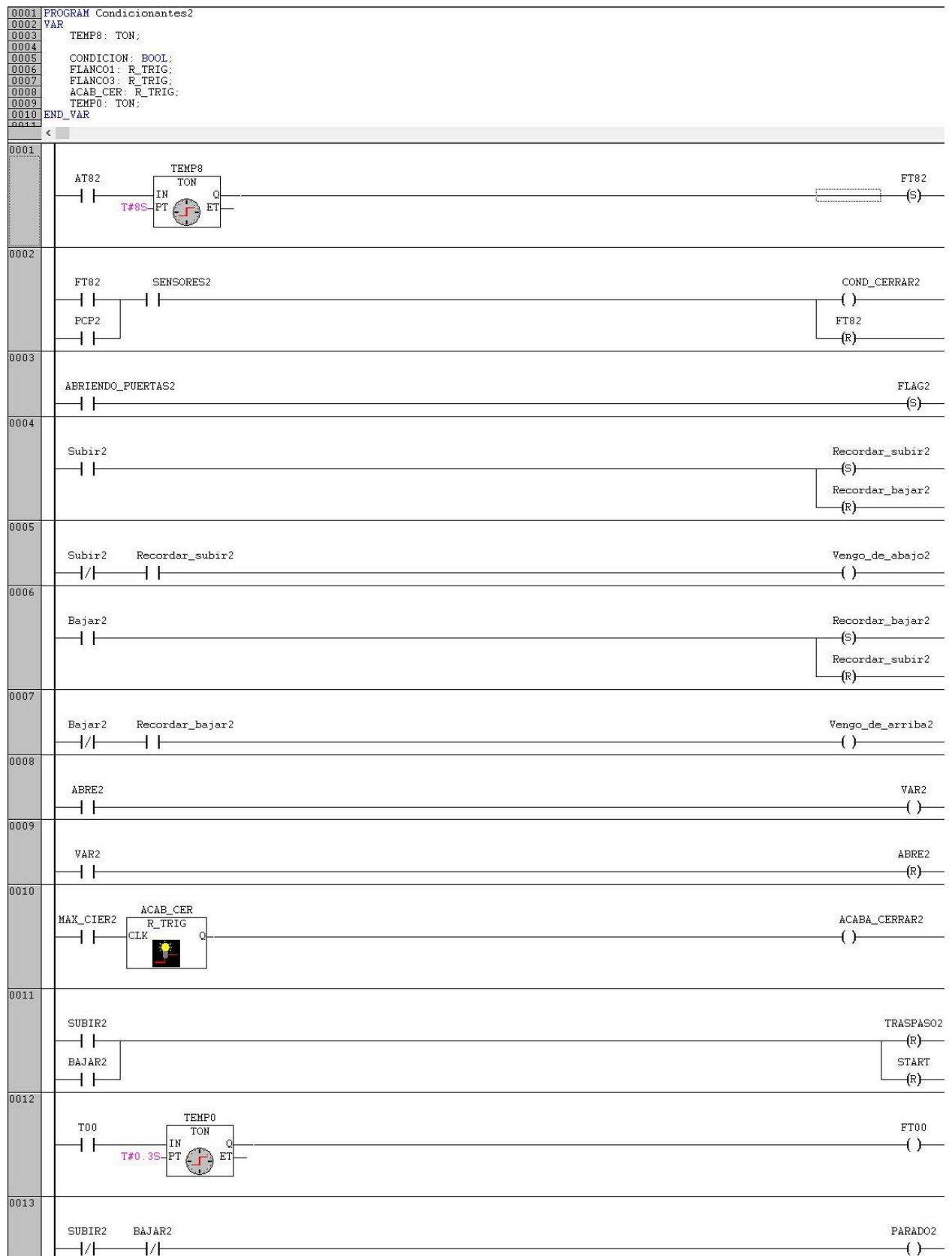
ELEVADOR 2 (SFC)



CONDICIONES 1 (LD)



CONDICIONES 2 (LD)



TURNOS (ST)

```

0001 PROGRAM nuevo
0002 VAR
0003   i: INT;
0004   CONT: INT;
0005   CONT2: INT;
0006   ALGO_PULSADO: INT;
0007   j: INT;
0008 END_VAR
0009 <
0010 (* ASCENSOR 1 Y 2 *)
0011 (*Elección del ascensor que atiende la llamada exterior de subida si están los dos activos*)
0012 IF STOP1=FALSE AND STOP2=FALSE THEN
0013   FOR i:=1 TO 10 BY 1 DO
0014     IF PULSADOR_SUB[i]=1 AND LUZ_SUB[i]=FALSE THEN
0015       ALGO_PULSADO:=i;
0016       LUZ_SUB[i]:=TRUE;
0017       (*Si ambos están vacíos, asistirá el más cercano si se encuentra parado o si ambos se están
0018       recolocando en sus plantas de reposo y la planta de reposo es más cercana al piso de destino*)
0019       IF OCUPADO=FALSE AND OCUPADO2=FALSE THEN
0020         IF (ABS(PLANTA2-ALGO_PULSADO) < ABS(PLANTA-ALGO_PULSADO) AND ((PARADO2=TRUE) OR (PARADO=FALSE AND ALGO_PULSADO>3)))
0021         OR (ABS(PLANTA-ALGO_PULSADO) <= ABS(PLANTA2-ALGO_PULSADO) AND PARADO=FALSE AND (PARADO2=TRUE OR ALGO_PULSADO>3)) THEN
0022           IF NO_ACEPTAMOS2=FALSE THEN
0023             PISO_PULSADO:=i;
0024             AUXILIAR2:=i;
0025             OCUPADO2:=TRUE;
0026             IF i<PLANTA2 THEN
0027               RECOGIDA_SUBIR2:=TRUE;
0028             END_IF
0029           END_IF
0030           TURNO2_SUB[i]:=1;
0031         ELSE
0032           IF NO_ACEPTAMOS=FALSE THEN
0033             PISO_PULSADO:=i;
0034             AUXILIAR:=i;
0035             OCUPADO:=TRUE;
0036             IF i<PLANTA THEN
0037               RECOGIDA_SUBIR:=TRUE;
0038             END_IF
0039           END_IF
0040           TURNO_SUB[i]:=1;
0041         END_IF
0042       (*Si sólo uno está libre y sin realizar apertura o cierre de puertas, asistirá el libre*)
0043       ELSIF OCUPADO=FALSE AND OCUPADO2=TRUE THEN
0044         IF NO_ACEPTAMOS=FALSE THEN
0045           PISO_PULSADO:=i;
0046           AUXILIAR:=i;
0047           OCUPADO:=TRUE;
0048           IF i<PLANTA THEN
0049             RECOGIDA_SUBIR:=TRUE;
0050           END_IF
0051         ELSE
0052           TURNO2_SUB[i]:=1;
0053           IF RECOGIDA_SUBIR2=TRUE AND i<PISO_PULSADO2 THEN
0054             PISO_PULSADO2:=i;
0055             AUXILIAR2:=i;
0056           END_IF
0057         END_IF
0058         TURNO_SUB[i]:=1;
0059       ELSIF OCUPADO=TRUE AND OCUPADO2=FALSE THEN
0060         IF NO_ACEPTAMOS2=FALSE THEN
0061           PISO_PULSADO2:=i;
0062           AUXILIAR2:=i;
0063           OCUPADO2:=TRUE;
0064           IF i<PLANTA2 THEN
0065             RECOGIDA_SUBIR2:=TRUE;
0066           END_IF
0067         ELSE
0068           TURNO2_SUB[i]:=1;
0069           IF RECOGIDA_SUBIR=TRUE AND i<PISO_PULSADO THEN
0070             PISO_PULSADO:=i;
0071             AUXILIAR:=i;
0072           END_IF
0073         END_IF
0074         TURNO2_SUB[i]:=1;
0075       ELSIF OCUPADO=TRUE AND OCUPADO2=TRUE THEN
0076         IF TURNO2_SUB[i]=0 THEN
0077           TURNO2_SUB[i]:=1;
0078           IF RECOGIDA_SUBIR=TRUE AND i<PISO_PULSADO THEN
0079             PISO_PULSADO:=i;
0080             AUXILIAR:=i;
0081           END_IF
0082         END_IF
0083         TURNO2_SUB[i]:=1;
0084       ELSE
0085         IF TURNO2_SUB[i]=0 THEN
0086           TURNO2_SUB[i]:=1;
0087           IF RECOGIDA_SUBIR2=TRUE AND i<PISO_PULSADO2 THEN
0088             PISO_PULSADO2:=i;
0089             AUXILIAR2:=i;
0090           END_IF
0091         END_IF
0092       END_IF
0093     END_IF
0094   END_FOR
0095   (*Elección del ascensor que atiende la llamada exterior de bajada si están los dos activos*)
0096   IF STOP1=FALSE AND STOP2=FALSE THEN
0097     FOR i:=1 TO 10 BY 1 DO
0098       IF PULSADOR_BAJ[i]=1 AND LUZ_BAJ[i]=FALSE THEN
0099         ALGO_PULSADO:=i;
0100         LUZ_BAJ[i]:=TRUE;
0101         (*Si ambos están vacíos, asistirá el más cercano si se encuentra parado o si ambos se están
0102         recolocando en sus plantas de reposo y la planta de reposo es más cercana al piso de destino*)
0103         IF OCUPADO=FALSE AND OCUPADO2=FALSE THEN
0104           IF (ABS(PLANTA2-ALGO_PULSADO) < ABS(PLANTA-ALGO_PULSADO) AND ((PARADO2=TRUE) OR (PARADO=FALSE AND ALGO_PULSADO>3)))
0105           OR (ABS(PLANTA-ALGO_PULSADO) <= ABS(PLANTA2-ALGO_PULSADO) AND PARADO=FALSE AND (PARADO2=TRUE OR ALGO_PULSADO>3)) THEN
0106             IF NO_ACEPTAMOS2=FALSE THEN
0107               PISO_PULSADO2:=i;
0108               AUXILIAR2:=i;
0109               OCUPADO2:=TRUE;
0110               IF i>PLANTA2 THEN
0111                 RECOGIDA_BAJAR2:=TRUE;
0112               END_IF
0113             END_IF
0114           ELSE
0115             IF NO_ACEPTAMOS=FALSE THEN
0116               PISO_PULSADO:=i;
0117               AUXILIAR:=i;
0118               OCUPADO:=TRUE;
0119               IF i>PLANTA THEN
0120                 RECOGIDA_BAJAR:=TRUE;
0121               END_IF
0122             END_IF
0123           END_IF
0124           TURNO2_SUB[i]:=1;
0125         ELSE
0126           IF NO_ACEPTAMOS2=FALSE THEN
0127             PISO_PULSADO2:=i;
0128             AUXILIAR2:=i;
0129             OCUPADO2:=TRUE;
0130             IF i>PLANTA2 THEN
0131               RECOGIDA_BAJAR2:=TRUE;
0132             END_IF
0133           END_IF
0134         END_IF
0135         TURNO2_SUB[i]:=1;
0136       ELSE
0137         IF NO_ACEPTAMOS=FALSE THEN
0138           PISO_PULSADO:=i;
0139           AUXILIAR:=i;
0140           OCUPADO:=TRUE;
0141           IF i>PLANTA THEN
0142             RECOGIDA_BAJAR:=TRUE;
0143           END_IF
0144         END_IF
0145         TURNO2_SUB[i]:=1;
0146       END_IF
0147     END_FOR
0148   END_IF
0149 END_PROGRAM

```

```

0106         END_IF
0107     END_IF
0108     TURNO2_BAJ[i]:=1;
0109 ELSE
0110     IF NO_ACEPTAMOS=FALSE THEN
0111         PISO_PULSADO:=i;
0112         AUXILIAR:=i;
0113         OCUPADO:=TRUE;
0114         IF i>PLANTA THEN
0115             RECOGIDA_BAJAR:=TRUE;
0116         END_IF
0117     END_IF
0118     TURNO_BAJ[i]:=1;
0119 END_IF
0120 (*Si sólo uno está libre y sin realizar apertura o cierre de puertas, asistirá el libre*)
0121 ELSIF OCUPADO=FALSE AND OCUPADO2=TRUE THEN
0122     IF NO_ACEPTAMOS=FALSE THEN
0123         PISO_PULSADO:=i;
0124         AUXILIAR:=i;
0125         OCUPADO:=TRUE;
0126         IF i>PLANTA THEN
0127             RECOGIDA_BAJAR:=TRUE;
0128         END_IF
0129     ELSE
0130         TURNO2_BAJ[i]:=1;
0131         IF RECOGIDA_BAJAR2=TRUE AND i>PISO_PULSADO2 THEN
0132             PISO_PULSADO2:=i;
0133             AUXILIAR2:=i;
0134         END_IF
0135     END_IF
0136     TURNO_BAJ[i]:=1;
0137 ELSIF OCUPADO=TRUE AND OCUPADO2=FALSE THEN
0138     IF NO_ACEPTAMOS2=FALSE THEN
0139         PISO_PULSADO2:=i;
0140         AUXILIAR2:=i;
0141         OCUPADO2:=TRUE;
0142         IF i>PLANTA2 THEN
0143             RECOGIDA_BAJAR2:=TRUE;
0144         END_IF
0145     ELSE
0146         TURNO_BAJ[i]:=1;
0147         IF RECOGIDA_BAJAR=TRUE AND i>PISO_PULSADO THEN
0148             PISO_PULSADO:=i;
0149             AUXILIAR:=i;
0150         END_IF
0151     END_IF
0152     TURNO2_BAJ[i]:=1;
0153 (*Si ambos están ocupados, la adquieren los dos*)
0154 ELSIF OCUPADO=TRUE AND OCUPADO2=TRUE THEN
0155     IF TURNO_BAJ[i]=0 THEN
0156         TURNO_BAJ[i]:=1;
0157         IF RECOGIDA_BAJAR=TRUE AND i>PISO_PULSADO THEN
0158             PISO_PULSADO:=i;
0159             AUXILIAR:=i;
0160         END_IF
0161     END_IF
0162     IF TURNO2_BAJ[i]=0 THEN
0163         TURNO2_BAJ[i]:=1;
0164         IF RECOGIDA_BAJAR2=TRUE AND i>PISO_PULSADO2 THEN
0165             PISO_PULSADO2:=i;
0166             AUXILIAR2:=i;
0167         END_IF
0168     END_IF
0169 END_IF
0170 END_IF
0171 END_FOR
0172 END_IF
0173
0174
0175 (* Si un ascensor tiene una llamada de destino distinto de la del otro y
0176 este otro no tiene una llamada interior al piso de destino del primero, la llamada la pierde*)
0177 FOR i:=1 TO 10 BY 1 DO
0178     IF PISO_PULSADO=i AND LUZ_INT2[i]=FALSE AND PISO_PULSADO2<i AND
0179     ((RECOGIDA_SUBIR=FALSE AND RECOGIDA_BAJAR=FALSE) OR ((RECOGIDA_SUBIR=TRUE OR RECOGIDA_BAJAR=TRUE) AND CONT2=1)) THEN
0180         TURNO2_SUB[i]:=FALSE;
0181         TURNO2_BAJ[i]:=FALSE;
0182     ELSIF PISO_PULSADO2=i AND LUZ_INT[i]=FALSE AND PISO_PULSADO<i AND
0183     ((RECOGIDA_SUBIR2=FALSE AND RECOGIDA_BAJAR2=FALSE) OR ((RECOGIDA_SUBIR2=TRUE OR RECOGIDA_BAJAR2=TRUE) AND CONT=1)) THEN
0184         TURNO_SUB[i]:=FALSE;
0185         TURNO_BAJ[i]:=FALSE;
0186     END_IF
0187 END_FOR
0188
0189
0190 (*Si ambos ascensores están en parada forzada todas las llamadas se anularán*)
0191 IF STOP1=TRUE AND STOP2=TRUE THEN
0192     FOR i:=10 TO 1 BY -1 DO
0193         TURNO_SUB[i]:=0;
0194         TURNO_BAJ[i]:=0;
0195         TURNO2_SUB[i]:=0;
0196         TURNO2_BAJ[i]:=0;
0197         LUZ_INT[i]:=0;
0198         LUZ_INT2[i]:=0;
0199         LUZ_SUB[i]:=0;
0200         LUZ_BAJ[i]:=0;
0201         OCUPADO:=FALSE;
0202         OCUPADO2:=FALSE;
0203     END_FOR
0204 END_IF
0205
0206
0207 (*ASCENSOR 1*)
0208
0209 (*Si se realizada llamada interior*)
0210 IF STOP1=FALSE THEN
0211     FOR i:=1 TO 10 BY 1 DO
0212         IF PULSADOR_INT1[i]=TRUE AND FLAG=TRUE AND LUZ_INT[i]=FALSE AND ((PARADO=TRUE AND PLANTA<i) OR PARADO=FALSE) THEN
0213             LUZ_INT[i]:=TRUE;
0214             (*Si la planta destino es inferior a la planta actual, será de bajada*)
0215             IF (i<PLANTA AND SUBIR=FALSE) OR (i<=PLANTA AND SUBIR=TRUE) THEN
0216                 TURNO_BAJ[i]:=1;
0217             (*Si la planta destino es superior a la actual será de subida*)
0218             ELSIF (i>PLANTA AND BAJAR=FALSE) OR (i>=PLANTA AND BAJAR=TRUE) THEN
0219                 TURNO_SUB[i]:=1;
0220             END_IF
0221             (*Si estaba libre, será planta de destino*)
0222             IF OCUPADO=FALSE AND NO_ACEPTAMOS=FALSE THEN
0223                 PISO_PULSADO:=i;
0224                 AUXILIAR:=i;
0225             END_IF
0226             OCUPADO:=TRUE;
0227         END_IF
0228     END_FOR
0229 END_IF

```



```

0230
0231
0232 (*Si la planta de destino es atendida, el último movimiento es de subida.
0233 no he atendido llamada para bajar y la planta actual no está accionada*)
0234 IF PISO_PULSADO<>0 THEN
0235 IF TURNO_SUB[PISO_PULSADO]=0 AND ACABA_CERRAR=TRUE AND Vengo_de_abajo=TRUE AND RECOGIDA_BAJAR=FALSE THEN
0236 IF LUZ_BAJ[PISO_PULSADO]=FALSE AND LUZ_SUB[PISO_PULSADO]=FALSE AND TURNO_BAJ[PISO_PULSADO]=FALSE THEN
0237 FOR i:=PISO_PULSADO TO 10 BY 1 DO
0238 (*Comprobamos si hay llamada de planta superior para subir*)
0239 IF TURNO_SUB[i]=1 AND (LUZ_INT[i]=TRUE OR PISO_PULSADO2<>i OR STOP2=TRUE) THEN
0240 PISO_PULSADO:=i;
0241 i:=11;
0242 END_IF
0243 END_FOR
0244 (*Si no hay, comprobamos planta para bajar*)
0245 IF AUXILIAR=PISO_PULSADO THEN
0246 FOR i:=10 TO 1 BY -1 DO
0247 IF TURNO_BAJ[i]=1 AND (LUZ_INT[i]=TRUE OR PISO_PULSADO2<>i OR STOP2=TRUE) THEN
0248 IF i>PISO_PULSADO THEN
0249 RECOGIDA_BAJAR:=TRUE;
0250 END_IF
0251 PISO_PULSADO:=i;
0252 i:=0;
0253 END_IF
0254 END_FOR
0255 (*Si no hay, comprobamos planta inferior para subir*)
0256 IF AUXILIAR=PISO_PULSADO THEN
0257 FOR i:=1 TO PISO_PULSADO BY 1 DO
0258 IF TURNO_SUB[i]=1 AND (LUZ_INT[i]=TRUE OR PISO_PULSADO2<>i OR STOP2=TRUE) THEN
0259 PISO_PULSADO:=i;
0260 RECOGIDA_SUBIR:=TRUE;
0261 i:=11;
0262 END_IF
0263 END_FOR
0264 IF AUXILIAR=PISO_PULSADO THEN
0265 PISO_PULSADO:=0;
0266 END_IF
0267 END_IF
0268 AUXILIAR:=PISO_PULSADO;
0269 END_IF
0270 END_IF
0271 END_IF
0272 END_IF
0273
0274
0275 (*Si la planta de destino es atendida, el último movimiento es de bajada.
0276 no he atendido llamada para subir y la planta actual no está accionada*)
0277 IF PISO_PULSADO<>0 THEN
0278 IF TURNO_BAJ[PISO_PULSADO]=0 AND ACABA_CERRAR=TRUE AND Vengo_de_arriba=TRUE AND RECOGIDA_SUBIR=FALSE THEN
0279 IF LUZ_BAJ[PISO_PULSADO]=FALSE AND LUZ_SUB[PISO_PULSADO]=FALSE AND TURNO_SUB[PISO_PULSADO]=FALSE THEN
0280 FOR i:=PISO_PULSADO TO 1 BY -1 DO
0281 (*Comprobamos si hay llamada de planta inferior para bajar*)
0282 IF TURNO_BAJ[i]=1 AND (LUZ_INT[i]=TRUE OR PISO_PULSADO2<>i OR STOP2=TRUE) THEN
0283 PISO_PULSADO:=i;
0284 i:=0;
0285 END_IF
0286 END_FOR
0287 (*Si no hay, comprobamos planta para subir*)
0288 IF AUXILIAR=PISO_PULSADO THEN
0289 FOR i:=1 TO 10 BY 1 DO
0290 IF TURNO_SUB[i]=1 AND (LUZ_INT[i]=TRUE OR PISO_PULSADO2<>i OR STOP2=TRUE) THEN
0291 IF i<PISO_PULSADO THEN
0292 RECOGIDA_SUBIR:=TRUE;
0293 END_IF
0294 PISO_PULSADO:=i;
0295 i:=11;
0296 END_IF
0297 END_FOR
0298 (*Si no hay comprobamos planta superior para bajar*)
0299 IF AUXILIAR=PISO_PULSADO THEN
0300 FOR i:=10 TO PISO_PULSADO BY -1 DO
0301 IF TURNO_BAJ[i]=1 AND (LUZ_INT[i]=TRUE OR PISO_PULSADO2<>i OR STOP2=TRUE) THEN
0302 PISO_PULSADO:=i;
0303 RECOGIDA_BAJAR:=TRUE;
0304 i:=0;
0305 END_IF
0306 END_FOR
0307 IF AUXILIAR=PISO_PULSADO THEN
0308 PISO_PULSADO:=0;
0309 END_IF
0310 END_IF
0311 AUXILIAR:=PISO_PULSADO;
0312 END_IF
0313 END_IF
0314 END_IF
0315 END_IF
0316
0317
0318 (*Si la planta de destino es atendida, el último movimiento es de bajada.
0319 he atendido llamada para subir y la planta actual no está accionada*)
0320 IF PISO_PULSADO<>0 THEN
0321 IF TURNO_SUB[PISO_PULSADO]=0 AND ACABA_CERRAR=TRUE AND Vengo_de_arriba=TRUE AND RECOGIDA_SUBIR=TRUE THEN
0322 IF LUZ_BAJ[PISO_PULSADO]=FALSE AND LUZ_SUB[PISO_PULSADO]=FALSE AND TURNO_BAJ[PISO_PULSADO]=FALSE THEN
0323 FOR i:=10 TO PISO_PULSADO BY -1 DO
0324 (*Compruebo si hay piso de subida accionado*)
0325 IF TURNO_SUB[i]=1 AND (LUZ_INT[i]=TRUE OR PISO_PULSADO2<>i OR STOP2=TRUE) THEN
0326 PISO_PULSADO:=i;
0327 RECOGIDA_SUBIR:=FALSE;
0328 i:=0;
0329 END_IF
0330 END_FOR
0331 (*Si no hay, comprobamos si hay piso inferior de bajada para mantener el mismo movimiento*)
0332 IF AUXILIAR=PISO_PULSADO THEN
0333 FOR i:=PISO_PULSADO TO 1 BY -1 DO
0334 IF TURNO_BAJ[i]=1 AND (LUZ_INT[i]=TRUE OR PISO_PULSADO2<>i OR STOP2=TRUE) THEN
0335 PISO_PULSADO:=i;
0336 RECOGIDA_SUBIR:=FALSE;
0337 i:=0;
0338 END_IF
0339 END_FOR
0340 (*Si no hay, comprobamos piso inferior de subida*)
0341 IF AUXILIAR=PISO_PULSADO THEN
0342 FOR i:=1 TO PISO_PULSADO BY 1 DO
0343 IF TURNO_SUB[i]=1 AND (LUZ_INT[i]=TRUE OR PISO_PULSADO2<>i OR STOP2=TRUE) THEN
0344 PISO_PULSADO:=i;
0345 i:=11;
0346 END_IF
0347 END_FOR
0348 (*Si no hay, comprobamos piso superior de bajada*)
0349 IF AUXILIAR=PISO_PULSADO THEN
0350 FOR i:=10 TO PISO_PULSADO BY -1 DO
0351 IF TURNO_BAJ[i]=1 AND (LUZ_INT[i]=TRUE OR PISO_PULSADO2<>i OR STOP2=TRUE) THEN
0352 PISO_PULSADO:=i;
0353 RECOGIDA_BAJAR:=TRUE;
0354 RECOGIDA_SUBIR:=FALSE;
0355 i:=0;
0356 END_IF
0357 END_FOR
0358 IF AUXILIAR=PISO_PULSADO THEN
0359 PISO_PULSADO:=0;
0360 RECOGIDA_SUBIR:=FALSE;
0361 END_IF
0362 END_IF
0363 END_IF
0364 AUXILIAR:=PISO_PULSADO;
0365 END_IF
0366 END_IF
0367 END_IF
0368 END_IF

```

```

0369
0370
0371 (*Si la planta de destino es atendida, el último movimiento es de subida,
0372 he atendido llamada para bajar y la planta actual no está accionada*)
0373 IF PISO_PULSADO<>0 THEN
0374     IF TURNO_BAJ[PISO_PULSADO]=0 AND ACABA_CERRAR=TRUE AND Vengo_de_abajo=TRUE AND RECOGIDA_BAJAR=TRUE THEN
0375         IF LUZ_BAJ[PISO_PULSADO]=FALSE AND LUZ_SUB[PISO_PULSADO]=FALSE AND TURNO_SUB[PISO_PULSADO]=FALSE THEN
0376             FOR i:=1 TO PISO_PULSADO BY 1 DO
0377                 (*Comprobamos piso inferior de bajada*)
0378                 IF TURNO_BAJ[i]=1 AND (LUZ_INT[i]=TRUE OR PISO_PULSADO2<>i OR STOP2=TRUE) THEN
0379                     PISO_PULSADO:=i;
0380                     RECOGIDA_BAJAR:=FALSE;
0381                     i:=11;
0382                 END_IF
0383             END_FOR
0384             (*Si no hay, comprobamos piso superior de subida para mantener el mismo movimiento de cabina*)
0385             IF AUXILIAR=PISO_PULSADO THEN
0386                 FOR i:=PISO_PULSADO TO 10 BY 1 DO
0387                     IF TURNO_SUB[i]=1 AND (LUZ_INT[i]=TRUE OR PISO_PULSADO2<>i OR STOP2=TRUE) THEN
0388                         PISO_PULSADO:=i;
0389                         RECOGIDA_BAJAR:=FALSE;
0390                         i:=11;
0391                     END_IF
0392                 END_FOR
0393                 (*Si no hay, comprobamos piso superior de bajada*)
0394                 IF AUXILIAR=PISO_PULSADO THEN
0395                     FOR i:=10 TO PISO_PULSADO BY -1 DO
0396                         IF TURNO_BAJ[i]=1 AND (LUZ_INT[i]=TRUE OR PISO_PULSADO2<>i OR STOP2=TRUE) THEN
0397                             PISO_PULSADO:=i;
0398                             i:=0;
0399                         END_IF
0400                     END_FOR
0401                     (*Si no hay, comprobamos piso inferior de subida*)
0402                     IF AUXILIAR=PISO_PULSADO THEN
0403                         FOR i:=1 TO PISO_PULSADO BY 1 DO
0404                             IF TURNO_SUB[i]=1 AND (LUZ_INT[i]=TRUE OR PISO_PULSADO2<>i OR STOP2=TRUE) THEN
0405                                 PISO_PULSADO:=i;
0406                                 RECOGIDA_SUBIR:=TRUE;
0407                                 RECOGIDA_BAJAR:=FALSE;
0408                                 i:=11;
0409                             END_IF
0410                         END_FOR
0411                         IF AUXILIAR=PISO_PULSADO THEN
0412                             PISO_PULSADO:=0;
0413                             RECOGIDA_BAJAR:=FALSE;
0414                         END_IF
0415                     END_IF
0416                 END_IF
0417             END_IF
0418             AUXILIAR:=PISO_PULSADO;
0419         END_IF
0420     END_IF
0421 END_IF
0422
0423 (*Apagar luces de piso cuando se abrimos sus puertas*)
0424 FOR i:=1 TO 10 BY 1 DO
0425     IF ABRIENDO_PUERTAS=TRUE AND SENSOR[i]=TRUE THEN
0426         LUZ_INT[i]:=FALSE;
0427         TURNO_SUB[i]:=0;
0428         LUZ_SUB[i]:=0;
0429         TURNO_BAJ[i]:=0;
0430         LUZ_BAJ[i]:=0;
0431     (*Si el piso de destino del otro ascensor es el mismo y no es interna,
0432 se le asigna otro para anularle la llamada*)
0433     IF PISO_PULSADO2=i AND LUZ_INT2[i]=FALSE THEN
0434         IF Subir2=TRUE AND RECOGIDA_BAJAR2=FALSE THEN
0435             FOR j:=10 TO PLANTA2 BY -1 DO
0436                 IF TURNO2_SUB[j]=1 AND j<>i THEN
0437                     TURNO2_SUB[PISO_PULSADO2]:=0;
0438                     TURNO2_BAJ[PISO_PULSADO2]:=0;
0439                     PISO_PULSADO2:=j;
0440                     j:=0;
0441                 END_IF
0442             END_FOR
0443         ELSEIF Bajar2=TRUE AND RECOGIDA_SUBIR2=FALSE THEN
0444             FOR j:=PLANTA2 TO 1 BY -1 DO
0445                 IF TURNO2_BAJ[j]=1 AND j<>i THEN
0446                     TURNO2_BAJ[PISO_PULSADO2]:=0;
0447                     TURNO2_SUB[PISO_PULSADO2]:=0;
0448                     PISO_PULSADO2:=j;
0449                     j:=0;
0450                 END_IF
0451             END_FOR
0452         END_IF
0453     ELSEIF (PISO_PULSADO2<>i OR (SUBIR2=FALSE AND BAJAR2=FALSE)) AND LUZ_INT2[i]=FALSE THEN
0454         IF TURNO_SUB[i]=0 THEN
0455             TURNO2_SUB[i]:=0;
0456         END_IF
0457         IF TURNO_BAJ[i]=0 THEN
0458             TURNO2_BAJ[i]:=0;
0459         END_IF
0460     END_IF
0461 END_IF
0462 END_FOR
0463
0464 (*Si la cabina llega a un piso con llamada admisible para abrir puertas, estas se abriran*)
0465 FOR i:=1 TO 10 BY 1 DO
0466     IF SENSOR[i]=TRUE AND RECOGIDA_SUBIR=FALSE AND RECOGIDA_BAJAR=FALSE
0467     AND ((SUBIR=TRUE AND TURNO_SUB[i]=1) OR (BAJAR=TRUE AND TURNO_BAJ[i]=1)) THEN
0468         ABRE:=TRUE;
0469     END_IF
0470 END_FOR
0471
0472 (*Comprobación del estado de la memoria*)
0473 IF SIEMPRE=TRUE THEN
0474     CONT:=0;
0475     FOR i:=1 TO 10 BY 1 DO
0476         IF TURNO_SUB[i]<>0 OR TURNO_BAJ[i]<>0 THEN
0477             CONT:=CONT+1;
0478         END_IF
0479     END_FOR
0480     IF CONT=0 THEN
0481         OCUPADO:=FALSE;
0482     ELSE
0483         OCUPADO:=TRUE;
0484     END_IF
0485 END_IF
0486
0487
0488
0489
0490

```



```

0491 (**ASCENSOR 2**)
0492 (*Tendrá las mismas características que el ascensor 1 pero codificados para el 2º ascensor*)
0493
0494 IF STOP2=FALSE THEN
0495   FOR i:=1 TO 10 BY 1 DO
0496     IF PULSADOR_INT2[i]=TRUE AND FLAG2=TRUE AND LUZ_INT2[i]=FALSE AND ((PARADO2=TRUE AND PLANTA2<>i) OR PARADO2=FALSE) THEN
0497       LUZ_INT2[i]:=TRUE;
0498       IF OCUPADO2=FALSE AND NO_ACEPTAMOS=FALSE THEN
0499         PISO_PULSADO2:=i;
0500         AUXILIAR2:=i;
0501       END_IF
0502       IF (i<PLANTA2 AND SUBIR2=FALSE) OR (i<=PLANTA2 AND SUBIR2=TRUE) THEN
0503         TURNO2_BAJ[i]:=1;
0504       ELSEIF (i>PLANTA2 AND BAJAR2=FALSE) OR (i>=PLANTA2 AND BAJAR2=TRUE) THEN
0505         TURNO2_SUB[i]:=1;
0506       END_IF
0507       OCUPADO2:=TRUE;
0508     END_IF
0509   END_FOR
0510 END_IF
0511 END_FOR
0512 END_IF
0513
0514 IF PISO_PULSADO2<>0 THEN
0515   IF TURNO2_SUB[PISO_PULSADO2]=0 AND ACABA_CERRAR2=TRUE AND Vengo_de_abajo2=TRUE AND RECOGIDA_BAJAR2=FALSE THEN
0516     IF LUZ_BAJ[PISO_PULSADO2]=FALSE AND LUZ_SUB[PISO_PULSADO2]=FALSE AND TURNO2_BAJ[PISO_PULSADO2]=FALSE THEN
0517       FOR i:=PISO_PULSADO2 TO 10 BY 1 DO
0518         IF TURNO2_SUB[i]=1 AND (LUZ_INT2[i]=TRUE OR PISO_PULSADO2<>i OR STOP1=TRUE) THEN
0519           PISO_PULSADO2:=i;
0520           i:=11;
0521         END_IF
0522       END_FOR
0523       IF AUXILIAR2=PISO_PULSADO2 THEN
0524         FOR i:=10 TO 1 BY -1 DO
0525           IF TURNO2_BAJ[i]=1 AND (LUZ_INT2[i]=TRUE OR PISO_PULSADO2<>i OR STOP1=TRUE) THEN
0526             IF i>PISO_PULSADO2 THEN
0527               RECOGIDA_BAJAR2:=TRUE;
0528             END_IF
0529             PISO_PULSADO2:=i;
0530             i:=0;
0531           END_IF
0532         END_FOR
0533         IF AUXILIAR2=PISO_PULSADO2 THEN
0534           FOR i:=1 TO PISO_PULSADO2 BY 1 DO
0535             IF TURNO2_SUB[i]=1 AND (LUZ_INT2[i]=TRUE OR PISO_PULSADO2<>i OR STOP1=TRUE) THEN
0536               PISO_PULSADO2:=i;
0537               RECOGIDA_SUBIR2:=TRUE;
0538               i:=11;
0539             END_IF
0540           END_FOR
0541           IF AUXILIAR2=PISO_PULSADO2 THEN
0542             PISO_PULSADO2:=0;
0543           END_IF
0544         END_IF
0545       END_IF
0546       AUXILIAR2:=PISO_PULSADO2;
0547     END_IF
0548   END_IF
0549 END_IF
0550
0551 IF PISO_PULSADO2<>0 THEN
0552   IF TURNO2_BAJ[PISO_PULSADO2]=0 AND ACABA_CERRAR2=TRUE AND Vengo_de_arriba2=TRUE AND RECOGIDA_SUBIR2=FALSE THEN
0553     IF LUZ_BAJ[PISO_PULSADO2]=FALSE AND LUZ_SUB[PISO_PULSADO2]=FALSE AND TURNO2_SUB[PISO_PULSADO2]=FALSE THEN
0554       FOR i:=PISO_PULSADO2 TO 1 BY -1 DO
0555         IF TURNO2_BAJ[i]=1 AND (LUZ_INT2[i]=TRUE OR PISO_PULSADO2<>i OR STOP1=TRUE) THEN
0556           PISO_PULSADO2:=i;
0557           i:=0;
0558         END_IF
0559       END_FOR
0560       IF AUXILIAR2=PISO_PULSADO2 THEN
0561         FOR i:=10 TO 1 BY -1 DO
0562           IF TURNO2_SUB[i]=1 AND (LUZ_INT2[i]=TRUE OR PISO_PULSADO2<>i OR STOP1=TRUE) THEN
0563             IF i<PISO_PULSADO2 THEN
0564               RECOGIDA_SUBIR2:=TRUE;
0565             END_IF
0566             PISO_PULSADO2:=i;
0567             i:=11;
0568           END_IF
0569         END_FOR
0570         IF AUXILIAR2=PISO_PULSADO2 THEN
0571           FOR i:=10 TO PISO_PULSADO2 BY -1 DO
0572             IF TURNO2_BAJ[i]=1 AND (LUZ_INT2[i]=TRUE OR PISO_PULSADO2<>i OR STOP1=TRUE) THEN
0573               PISO_PULSADO2:=i;
0574               RECOGIDA_BAJAR2:=TRUE;
0575               i:=0;
0576             END_IF
0577           END_FOR
0578           IF AUXILIAR2=PISO_PULSADO2 THEN
0579             PISO_PULSADO2:=0;
0580           END_IF
0581         END_IF
0582       END_IF
0583       AUXILIAR2:=PISO_PULSADO2;
0584     END_IF
0585   END_IF
0586 END_IF
0587 END_IF
0588
0589 IF PISO_PULSADO2<>0 THEN
0590   IF TURNO2_SUB[PISO_PULSADO2]=0 AND ACABA_CERRAR2=TRUE AND Vengo_de_arriba2=TRUE AND RECOGIDA_SUBIR2=TRUE THEN
0591     IF LUZ_BAJ[PISO_PULSADO2]=FALSE AND LUZ_SUB[PISO_PULSADO2]=FALSE AND TURNO2_BAJ[PISO_PULSADO2]=FALSE THEN
0592       FOR i:=PISO_PULSADO2 TO 1 BY -1 DO
0593         IF TURNO2_SUB[i]=1 AND (LUZ_INT2[i]=TRUE OR PISO_PULSADO2<>i OR STOP1=TRUE) THEN
0594           PISO_PULSADO2:=i;
0595           RECOGIDA_SUBIR2:=FALSE;
0596           i:=0;
0597         END_IF
0598       END_FOR
0599       IF AUXILIAR2=PISO_PULSADO2 THEN
0600         FOR i:=PISO_PULSADO2 TO 1 BY -1 DO
0601           IF TURNO2_BAJ[i]=1 AND (LUZ_INT2[i]=TRUE OR PISO_PULSADO2<>i OR STOP1=TRUE) THEN
0602             PISO_PULSADO2:=i;
0603             RECOGIDA_SUBIR2:=FALSE;
0604             i:=0;
0605           END_IF
0606         END_FOR
0607         IF AUXILIAR2=PISO_PULSADO2 THEN
0608           FOR i:=1 TO PISO_PULSADO2 BY 1 DO
0609             IF TURNO2_SUB[i]=1 AND (LUZ_INT2[i]=TRUE OR PISO_PULSADO2<>i OR STOP1=TRUE) THEN
0610               PISO_PULSADO2:=i;
0611               i:=11;
0612             END_IF
0613           END_FOR
0614           IF AUXILIAR2=PISO_PULSADO2 THEN
0615             FOR i:=10 TO PISO_PULSADO2 BY -1 DO
0616               IF TURNO2_BAJ[i]=1 AND (LUZ_INT2[i]=TRUE OR PISO_PULSADO2<>i OR STOP1=TRUE) THEN
0617                 PISO_PULSADO2:=i;
0618                 RECOGIDA_BAJAR2:=TRUE;
0619                 RECOGIDA_SUBIR2:=FALSE;
0620                 i:=0;
0621               END_IF
0622             END_FOR
0623             IF AUXILIAR2=PISO_PULSADO2 THEN
0624               PISO_PULSADO2:=0;
0625             END_IF
0626           END_IF
0627         END_IF
0628       END_IF
0629     END_IF
0630   END_IF
0631   AUXILIAR2:=PISO_PULSADO2;
0632 END_IF
0633 END_IF
0634 END_IF

```



```

0635
0636
0637 IF PISO_PULSADO2<>0 THEN
0638   IF TURNO2_BAJ[PISO_PULSADO2]=0 AND ACABA_CERRAR2=TRUE AND Vengo_de_abajo2=TRUE AND RECOGIDA_BAJAR2=TRUE THEN
0639     IF LUZ_BAJ[PISO_PULSADO2]=FALSE AND LUZ_SUB[PISO_PULSADO2]=FALSE AND TURNO2_SUB[PISO_PULSADO2]=FALSE THEN
0640       FOR i:=1 TO PISO_PULSADO2 BY 1 DO
0641         IF TURNO2_BAJ[i]=1 AND (LUZ_INT2[i]=TRUE OR PISO_PULSADO<>i OR STOP1=TRUE) THEN
0642           PISO_PULSADO2:=i;
0643           RECOGIDA_BAJAR2:=FALSE;
0644           i:=11;
0645         END_IF
0646       END_FOR
0647     IF AUXILIAR2=PISO_PULSADO2 THEN
0648       FOR i:=PISO_PULSADO2 TO 10 BY 1 DO
0649         IF TURNO2_SUB[i]=1 AND (LUZ_INT2[i]=TRUE OR PISO_PULSADO<>i OR STOP1=TRUE) THEN
0650           PISO_PULSADO2:=i;
0651           RECOGIDA_BAJAR2:=FALSE;
0652           i:=11;
0653         END_IF
0654       END_FOR
0655     IF AUXILIAR2=PISO_PULSADO2 THEN
0656       FOR i:=10 TO PISO_PULSADO2 BY -1 DO
0657         IF TURNO2_BAJ[i]=1 AND (LUZ_INT2[i]=TRUE OR PISO_PULSADO<>i OR STOP1=TRUE) THEN
0658           PISO_PULSADO2:=i;
0659           i:=0;
0660         END_IF
0661       END_FOR
0662     IF AUXILIAR2=PISO_PULSADO2 THEN
0663       FOR i:=1 TO PISO_PULSADO2 BY 1 DO
0664         IF TURNO2_SUB[i]=1 AND (LUZ_INT2[i]=TRUE OR PISO_PULSADO<>i OR STOP1=TRUE) THEN
0665           PISO_PULSADO2:=i;
0666           RECOGIDA_SUBIR2:=TRUE;
0667           RECOGIDA_BAJAR2:=FALSE;
0668           i:=11;
0669         END_IF
0670       END_FOR
0671     IF AUXILIAR2=PISO_PULSADO2 THEN
0672       PISO_PULSADO2:=0;
0673       RECOGIDA_BAJAR2:=FALSE;
0674     END_IF
0675   END_IF
0676 END_IF
0677 END_IF
0678 AUXILIAR2:=PISO_PULSADO2;
0679 END_IF
0680 END_IF
0681 END_IF
0682
0683
0684 FOR i:=1 TO 10 BY 1 DO
0685   IF ABRIENDO_PUERTAS2=TRUE AND SENSOR2[i]=TRUE THEN
0686     LUZ_INT2[i]:=FALSE;
0687     TURNO2_BAJ[i]:=0;
0688     LUZ_BAJ[i]:=0;
0689     TURNO2_SUB[i]:=0;
0690     LUZ_SUB[i]:=0;
0691     IF PISO_PULSADO=i AND LUZ_INT[i]=FALSE THEN
0692       IF Subir=TRUE AND RECOGIDA_BAJAR=FALSE THEN
0693         FOR j:=10 TO PLANTA BY -1 DO
0694           IF TURNO2_SUB[j]=1 AND j<>i THEN
0695             TURNO2_SUB[PISO_PULSADO]:=0;
0696             PISO_PULSADO:=j;
0697             j:=0;
0698           END_IF
0699         END_FOR
0700       ELSEIF Bajar=TRUE AND RECOGIDA_SUBIR=FALSE THEN
0701         FOR j:=PLANTA TO 1 BY -1 DO
0702           IF TURNO2_BAJ[j]=1 AND j<>i THEN
0703             TURNO2_BAJ[PISO_PULSADO]:=0;
0704             PISO_PULSADO:=j;
0705             j:=0;
0706           END_IF
0707         END_FOR
0708       END_IF
0709     ELSEIF (PISO_PULSADO<>i OR (SUBIR=FALSE AND BAJAR=FALSE)) AND LUZ_INT[i]=FALSE THEN
0710       IF TURNO2_SUB[i]=0 THEN
0711         TURNO2_SUB[i]:=0;
0712       END_IF
0713       IF TURNO2_BAJ[i]=0 THEN
0714         TURNO2_BAJ[i]:=0;
0715       END_IF
0716     END_IF
0717   END_IF
0718 END_FOR
0719
0720
0721 FOR i:=1 TO 10 BY 1 DO
0722   IF SENSOR2[i]=TRUE AND RECOGIDA_SUBIR2=FALSE AND RECOGIDA_BAJAR2=FALSE AND
0723     ((SUBIR2=TRUE AND TURNO2_SUB[i]=1) OR (BAJAR2=TRUE AND TURNO2_BAJ[i]=1)) THEN
0724     ABRE2:=TRUE;
0725   END_IF
0726 END_FOR
0727
0728
0729 IF SIEMPRE=TRUE THEN
0730   CONT2:=0;
0731   FOR i:=1 TO 10 BY 1 DO
0732     IF TURNO2_SUB[i]<>0 OR TURNO2_BAJ[i]<>0 THEN
0733       CONT2:=CONT2+1;
0734     END_IF
0735   END_FOR
0736   IF CONT2=0 THEN
0737     OCUPADO2:=FALSE;
0738   ELSE
0739     OCUPADO2:=TRUE;
0740   END_IF
0741 END_IF

```

PARADAS (ST)

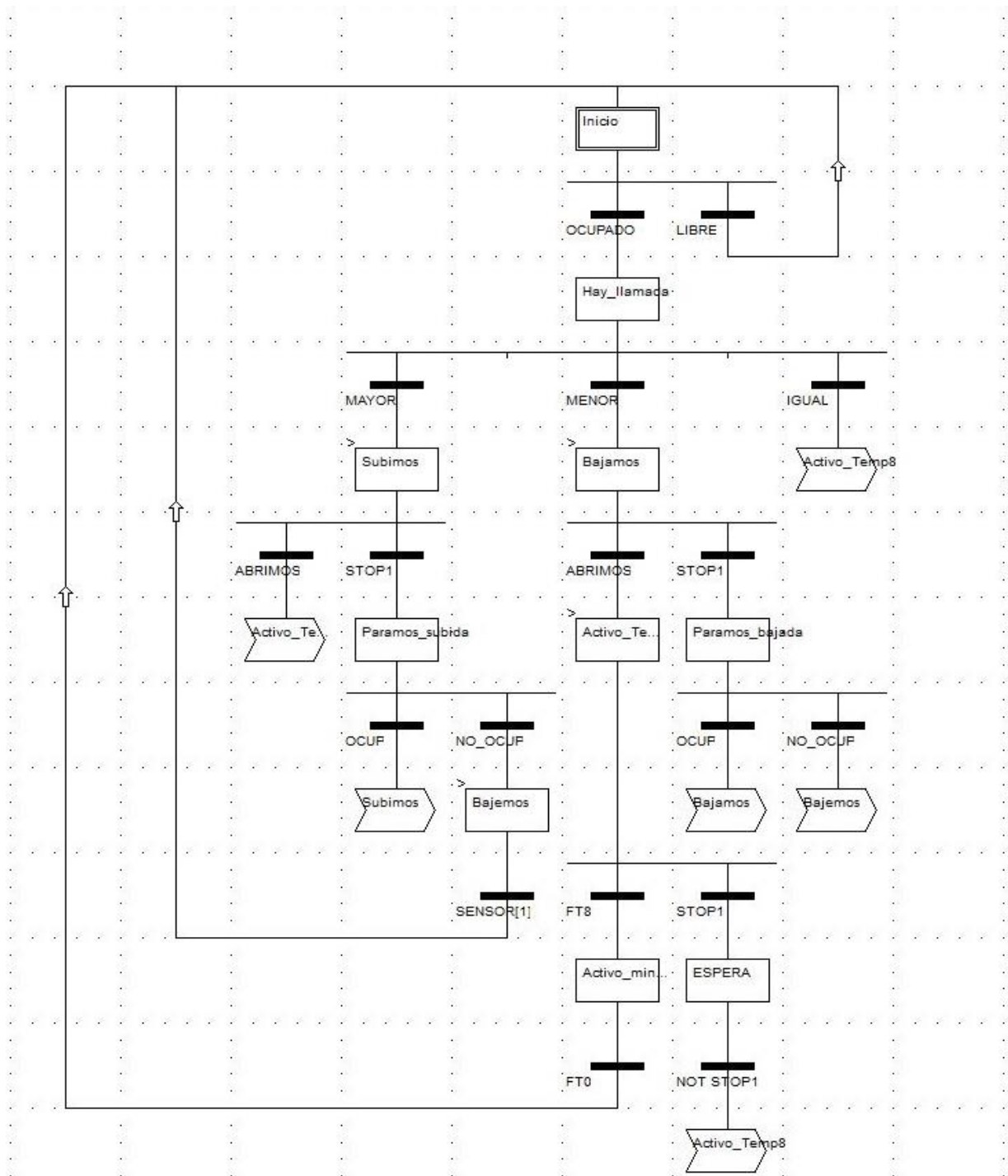
```

0001 PROGRAM PARADAS
0002 VAR
0003     i: INT;
0004     CONT: INT;
0005     CUENTA2: INT;
0006     CONT2: INT;
0007     ALGO_PULSADO: INT;
0008     j: INT;
0009 END_VAR
0010
0011 (*Si alguno de los dos ascensores se encuentra parado, todas las llamadas posteriores serán asignadas al ascensor en activo
0012 Si se mantiene parado más de 10 segundos, el ascensor parado cederá todas las llamadas pendientes del exterior*)
0013 IF STOP1=TRUE AND STOP2=FALSE THEN
0014     FOR i:=1 TO 10 BY 1 DO
0015         IF TURNO2_SUB[i]=0 AND ((PULSADOR_SUB[i]=1 AND LUZ_SUB[i]=FALSE) OR (LUZ_SUB[i]=TRUE AND FT10=TRUE)) THEN
0016             IF OCUPADO2=FALSE THEN
0017                 PISO_PULSADO2:=i;
0018                 AUXILIAR2:=i;
0019                 OCUPADO2:=TRUE;
0020                 IF i<PLANTA2 THEN
0021                     RECOGIDA_SUBIR2:=TRUE;
0022                 END_IF
0023             END_IF
0024             LUZ_SUB[i]:=TRUE;
0025             TURNO2_SUB[i]:=1;
0026             TURNO2_SUB[i]:=0;
0027         END_IF
0028         IF TURNO2_BAJ[i]=0 AND ((PULSADOR_BAJ[i]=1 AND LUZ_BAJ[i]=FALSE) OR (LUZ_BAJ[i]=TRUE AND FT10=TRUE)) THEN
0029             IF OCUPADO2=FALSE THEN
0030                 PISO_PULSADO2:=i;
0031                 AUXILIAR2:=i;
0032                 OCUPADO2:=TRUE;
0033                 IF i>PLANTA2 THEN
0034                     RECOGIDA_BAJAR2:=TRUE;
0035                 END_IF
0036             END_IF
0037             LUZ_BAJ[i]:=TRUE;
0038             TURNO2_BAJ[i]:=1;
0039             TURNO2_BAJ[i]:=0;
0040         END_IF
0041     END_FOR
0042 END_IF
0043
0044 IF STOP2=TRUE AND STOP1=FALSE THEN
0045     FOR i:=1 TO 10 BY 1 DO
0046         IF TURNO2_SUB[i]=0 AND ((PULSADOR_SUB[i]=1 AND LUZ_SUB[i]=FALSE) OR (LUZ_SUB[i]=TRUE AND FT10=TRUE)) THEN
0047             IF OCUPADO2=FALSE THEN
0048                 PISO_PULSADO:=i;
0049                 AUXILIAR:=i;
0050                 OCUPADO:=TRUE;
0051                 IF i<PLANTA THEN
0052                     RECOGIDA_SUBIR:=TRUE;
0053                 END_IF
0054             END_IF
0055             LUZ_SUB[i]:=TRUE;
0056             TURNO2_SUB[i]:=1;
0057             TURNO2_SUB[i]:=0;
0058         END_IF
0059         IF TURNO2_BAJ[i]=0 AND ((PULSADOR_BAJ[i]=1 AND LUZ_BAJ[i]=FALSE) OR (LUZ_BAJ[i]=TRUE AND FT10=TRUE)) THEN
0060             IF OCUPADO2=FALSE THEN
0061                 PISO_PULSADO:=i;
0062                 AUXILIAR:=i;
0063                 OCUPADO:=TRUE;
0064                 IF i>PLANTA THEN
0065                     RECOGIDA_BAJAR:=TRUE;
0066                 END_IF
0067             END_IF
0068             LUZ_BAJ[i]:=TRUE;
0069             TURNO2_BAJ[i]:=1;
0070             TURNO2_BAJ[i]:=0;
0071         END_IF
0072     END_FOR
0073 END_IF
0074
0075 (*Si estan vacios >40 seg. se recolocarán en sus pisos de reposo*)
0076 IF AMBOS_VACIOS=TRUE AND SENSOR[1]=FALSE THEN
0077     RECOLOCA:=TRUE;
0078     TURNO_BAJ[1]:=1;
0079     PISO_PULSADO:=1;
0080     OCUPADO:=TRUE;
0081     AUXILIAR:=1;
0082 END_IF
0083
0084 IF AMBOS_VACIOS=TRUE AND SENSOR2[5]=FALSE THEN
0085     RECOLOCA2:=TRUE;
0086     IF PLANTA2<5 THEN
0087         TURNO2_SUB[5]:=1;
0088         PISO_PULSADO2:=5;
0089         OCUPADO2:=TRUE;
0090         AUXILIAR2:=5;
0091     ELSE
0092         TURNO2_BAJ[5]:=1;
0093         PISO_PULSADO2:=5;
0094         OCUPADO2:=TRUE;
0095         AUXILIAR2:=5;
0096     END_IF
0097 END_IF
0098
0099 (*Si atienden una llamada en su trayecto hacia el piso de reposo, la llamada del piso de reposo desaparece*)
0100 IF ACABA_CERRAR2=TRUE AND RECOLOCA2=TRUE THEN
0101     RECOLOCA2:=FALSE;
0102     IF LUZ_INT2[5]=0 AND LUZ_SUB[5]=0 AND LUZ_BAJ[5]=0 THEN
0103         TURNO2_SUB[5]:=0;
0104         TURNO2_BAJ[5]:=0;
0105     END_IF
0106 END_IF
0107
0108 IF ACABA_CERRAR=TRUE AND RECOLOCA=TRUE THEN
0109     RECOLOCA:=FALSE;
0110     IF LUZ_INT[1]=0 THEN
0111         TURNO_BAJ[1]:=0;
0112     END_IF
0113 END_IF

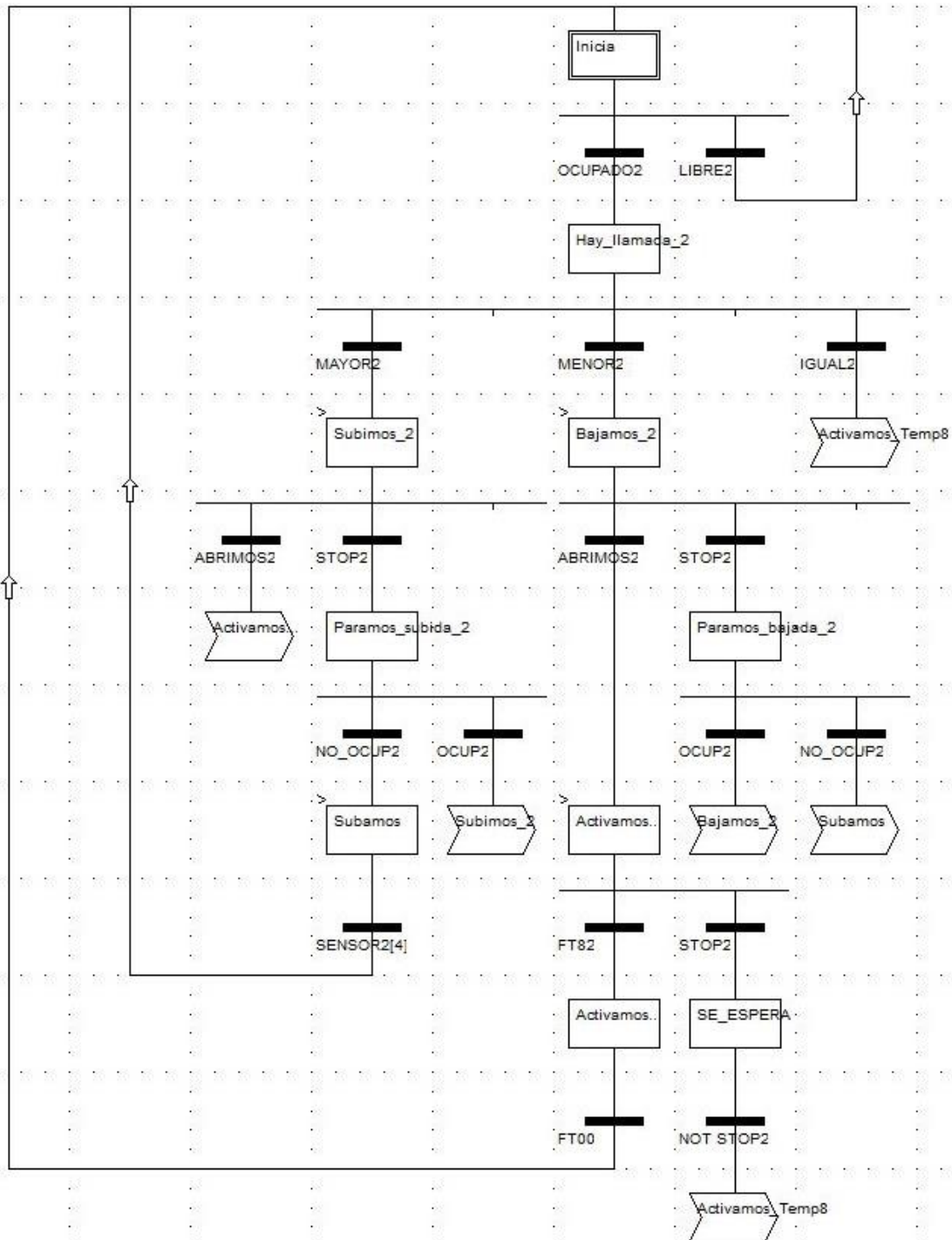
```

ANEXO IV: MAQUETA MANIOBRA COLECTIVA SELECTIVA

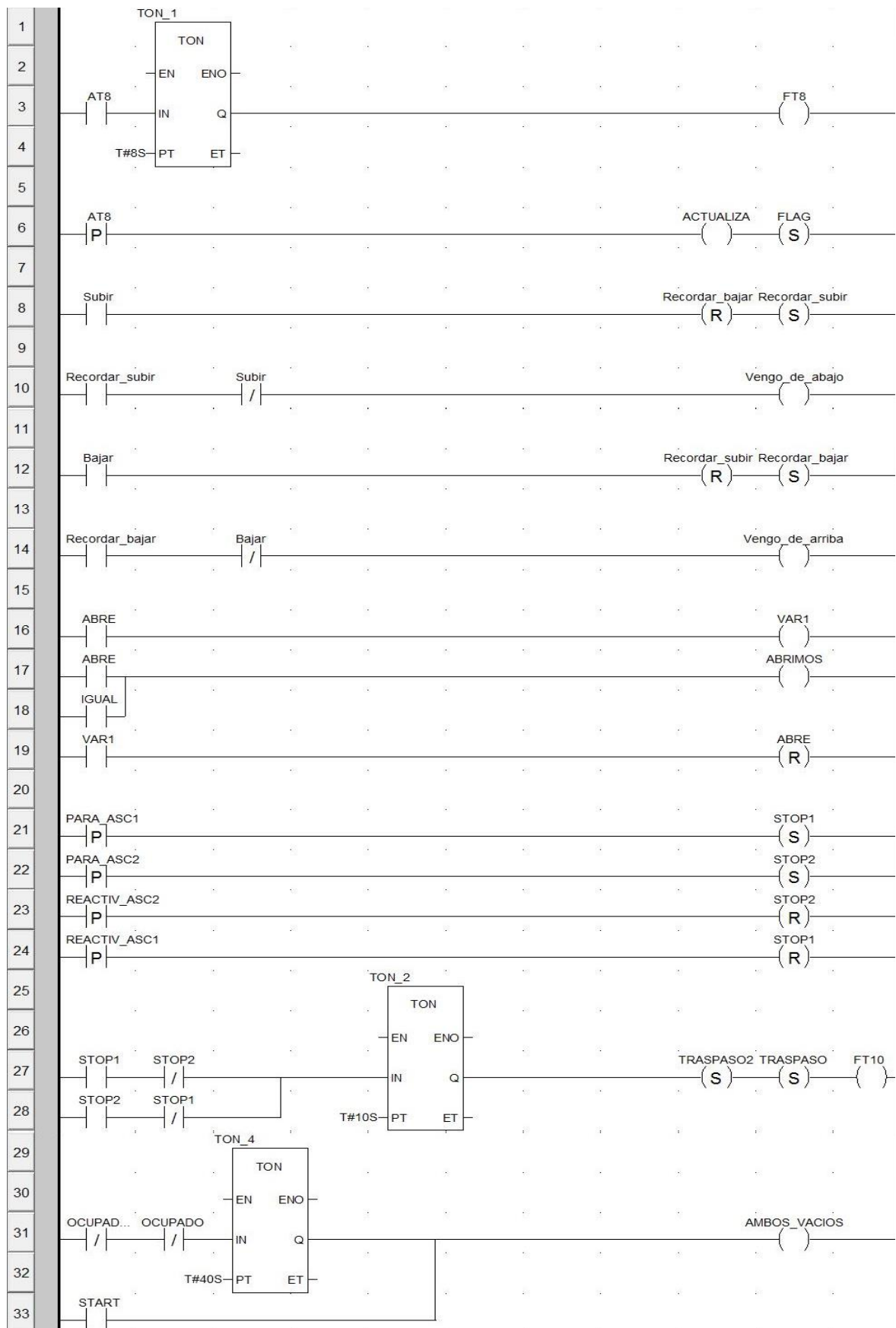
ASCENSOR 1 (SFC)



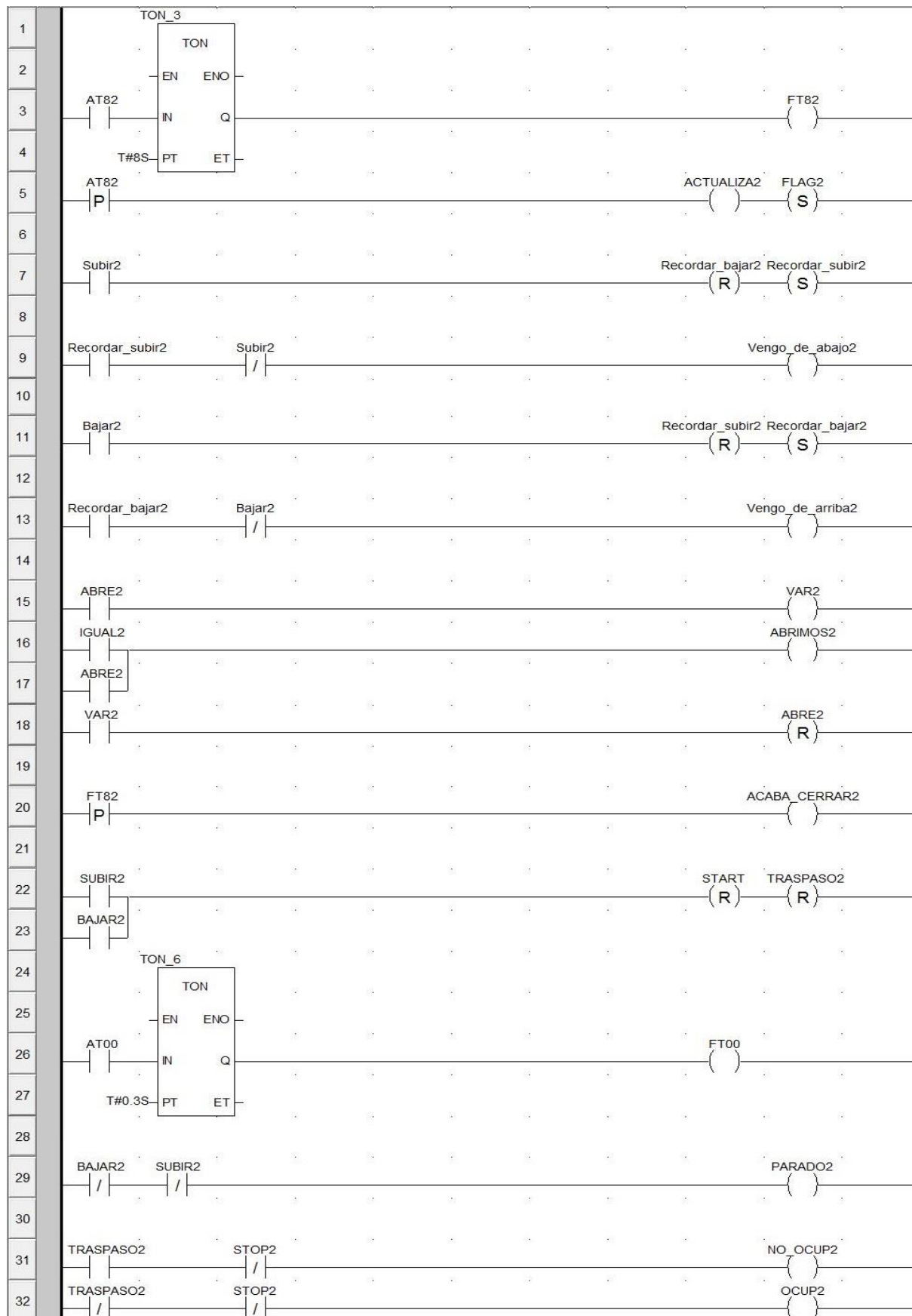
ASCENSOR 2 (SFC)



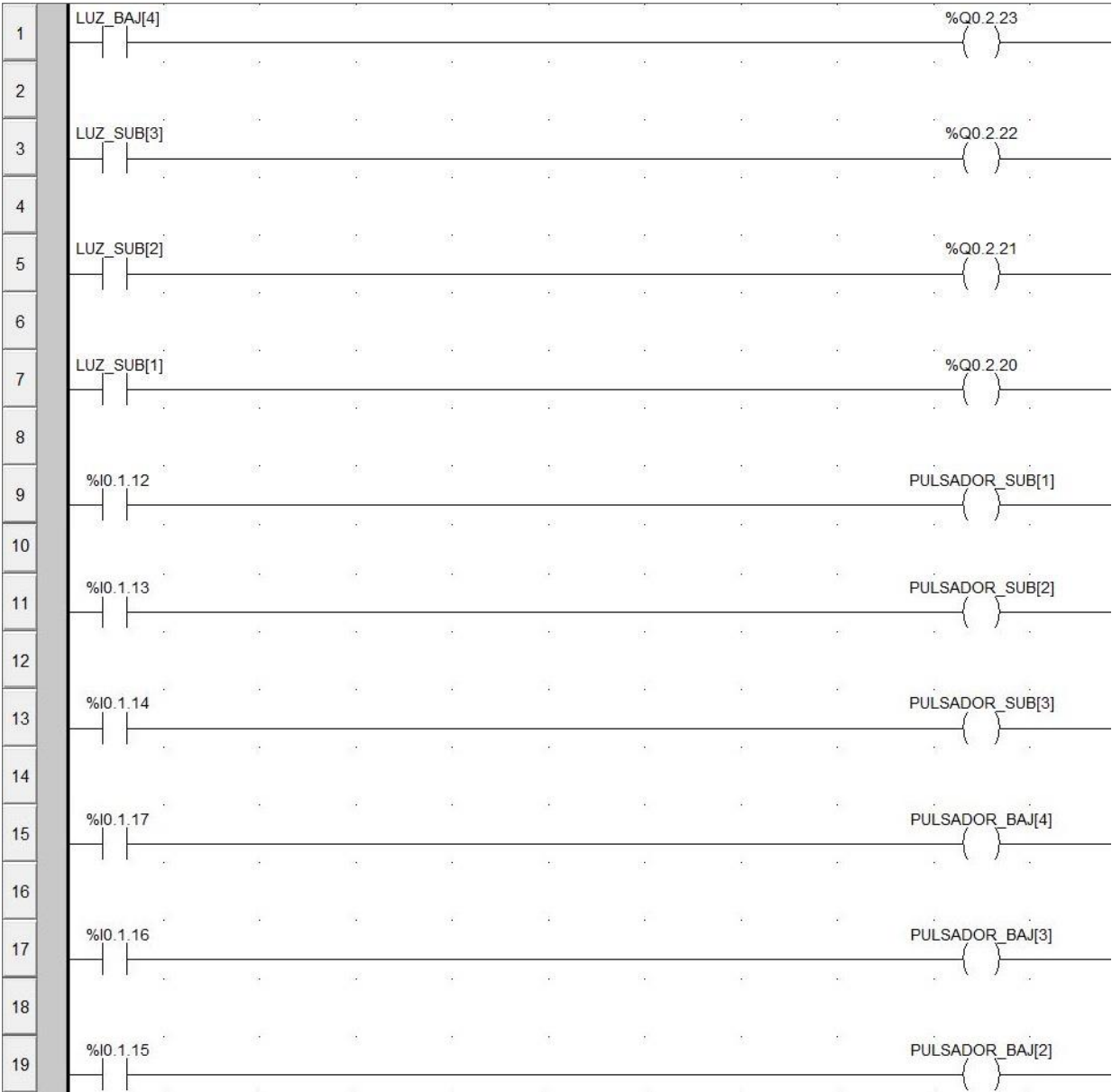
CONDICIONES 1 (LD)



CONDICIONES 2 (LD)



SALIDAS (LD)



TURNOS (ST)

```

(*COMUN*)

IF STOP1=FALSE AND STOP2=FALSE THEN
  FOR i:=1 TO 4 BY 1 DO
    IF PULSADOR_SUB[i]=1 AND LUZ_SUB[i]=FALSE THEN
      ALGO_PULSADO:=i;
      LUZ_SUB[i]:=TRUE;
      IF OCUPADO=FALSE AND OCUPADO2=FALSE THEN
        IF (ABS(PLANTA2-ALGO_PULSADO) < ABS(PLANTA-ALGO_PULSADO) AND ((PARADO2=TRUE) OR
          (PARADO=FALSE AND ALGO_PULSADO>2))) OR (ABS(PLANTA-ALGO_PULSADO) <= ABS(PLANTA2-ALGO_PULSADO)
          AND PARADO=FALSE AND (PARADO2=TRUE OR ALGO_PULSADO>2)) THEN
          IF AT82=FALSE THEN
            PISO_PULSADO2:=i;
            AUXILIAR2:=i;
            OCUPADO2:=TRUE;
            IF i<PLANTA2 THEN
              RECOGIDA_SUBIR2:=TRUE;
            END_IF;
          END_IF;
          TURNO2_SUB[i]:=1;
        ELSE
          IF AT8=FALSE THEN
            PISO_PULSADO:=i;
            AUXILIAR:=i;
            OCUPADO:=TRUE;
            IF i<PLANTA THEN
              RECOGIDA_SUBIR:=TRUE;
            END_IF;
          END_IF;
          TURNO_SUB[i]:=1;
        END_IF;
      ELSIF OCUPADO=FALSE AND OCUPADO2=TRUE THEN
        IF AT8=FALSE THEN
          PISO_PULSADO:=i;
          AUXILIAR:=i;
          OCUPADO:=TRUE;
          IF i<PLANTA THEN
            RECOGIDA_SUBIR:=TRUE;
          END_IF;
        ELSE
          TURNO2_SUB[i]:=1;
          IF RECOGIDA_SUBIR2=TRUE AND i<PISO_PULSADO2 THEN
            PISO_PULSADO2:=i;
            AUXILIAR2:=i;
          END_IF;
        END_IF;
        TURNO_SUB[i]:=1;
      ELSIF OCUPADO=TRUE AND OCUPADO2=FALSE THEN
        IF AT82=FALSE THEN
          PISO_PULSADO2:=i;
          AUXILIAR2:=i;
          OCUPADO2:=TRUE;
          IF i<PLANTA2 THEN
            RECOGIDA_SUBIR2:=TRUE;
          END_IF;
        ELSE
          TURNO_SUB[i]:=1;
          IF RECOGIDA_SUBIR=TRUE AND i<PISO_PULSADO THEN
            PISO_PULSADO:=i;
            AUXILIAR:=i;
          END_IF;
        END_IF;
        TURNO2_SUB[i]:=1;
      ELSIF OCUPADO=TRUE AND OCUPADO2=TRUE THEN
        IF TURNO_SUB[i]=0 THEN
          TURNO_SUB[i]:=1;
          IF RECOGIDA_SUBIR=TRUE AND i<PISO_PULSADO THEN
            PISO_PULSADO:=i;
            AUXILIAR:=i;
          END_IF;
        END_IF;
        IF TURNO2_SUB[i]=0 THEN
          TURNO2_SUB[i]:=1;
          IF RECOGIDA_SUBIR2=TRUE AND i<PISO_PULSADO2 THEN
            PISO_PULSADO2:=i;
            AUXILIAR2:=i;
          END_IF;
        END_IF;
      END_IF;
    END_IF;
  END_FOR;
END_IF;

IF STOP1=FALSE AND STOP2=FALSE THEN
  FOR i:=1 TO 4 BY 1 DO
    IF PULSADOR_BAJ[i]=1 AND LUZ_BAJ[i]=FALSE THEN
      ALGO_PULSADO:=i;
      LUZ_BAJ[i]:=TRUE;
      IF OCUPADO=FALSE AND OCUPADO2=FALSE THEN
        IF (ABS(PLANTA2-ALGO_PULSADO) < ABS(PLANTA-ALGO_PULSADO) AND ((PARADO2=TRUE) OR
          (PARADO=FALSE AND ALGO_PULSADO>2))) OR (ABS(PLANTA-ALGO_PULSADO) <= ABS(PLANTA2-ALGO_PULSADO)
          AND PARADO=FALSE AND (PARADO2=TRUE OR ALGO_PULSADO>2)) THEN
          IF AT82=FALSE THEN

```

```

AND PARADO=False AND (PARADO2=True OR ALGO_PULSADO>2)) THEN
  IF AT8=False THEN
    PISO_PULSADO2:=i;
    AUXILIAR2:=i;
    OCUPADO2:=True;
    IF i>PLANTA2 THEN
      RECOGIDA_BAJAR2:=True;
    END_IF;
  END_IF;
  TURNO2_BAJ[i]:=1;
ELSE
  IF AT8=False THEN
    PISO_PULSADO:=i;
    AUXILIAR:=i;
    OCUPADO:=True;
    IF i>PLANTA THEN
      RECOGIDA_BAJAR:=True;
    END_IF;
  END_IF;
  TURNO_BAJ[i]:=1;
END_IF;
ELSIF OCUPADO=False AND OCUPADO2=True THEN
  IF AT8=False THEN
    PISO_PULSADO:=i;
    AUXILIAR:=i;
    OCUPADO:=True;
    IF i>PLANTA THEN
      RECOGIDA_BAJAR:=True;
    END_IF;
  END_IF;
ELSE
  TURNO2_BAJ[i]:=1;
  IF RECOGIDA_BAJAR2=True AND i>PISO_PULSADO2 THEN
    PISO_PULSADO2:=i;
    AUXILIAR2:=i;
  END_IF;
END_IF;
  TURNO_BAJ[i]:=1;
ELSIF OCUPADO=True AND OCUPADO2=False THEN
  IF AT8=False THEN
    PISO_PULSADO2:=i;
    AUXILIAR2:=i;
    OCUPADO2:=True;
    IF i>PLANTA2 THEN
      RECOGIDA_BAJAR2:=True;
    END_IF;
  END_IF;
ELSE
  TURNO_BAJ[i]:=1;
  IF RECOGIDA_BAJAR=True AND i>PISO_PULSADO THEN
    PISO_PULSADO:=i;
    AUXILIAR:=i;
  END_IF;
END_IF;
  TURNO2_BAJ[i]:=1;
ELSIF OCUPADO=True AND OCUPADO2=True THEN
  IF TURNO_BAJ[i]=0 THEN
    TURNO_BAJ[i]:=1;
    IF RECOGIDA_BAJAR=True AND i>PISO_PULSADO THEN
      PISO_PULSADO:=i;
      AUXILIAR:=i;
    END_IF;
  END_IF;
  IF TURNO2_BAJ[i]=0 THEN
    TURNO2_BAJ[i]:=1;
    IF RECOGIDA_BAJAR2=True AND i>PISO_PULSADO2 THEN
      PISO_PULSADO2:=i;
      AUXILIAR2:=i;
    END_IF;
  END_IF;
END_IF;
END_IF;
END_FOR;
END_IF;

(*ASCENSOR 1*)
IF STOP1=False THEN
  FOR i:=1 TO 4 BY 1 DO
    IF PULSADOR_INT1[i]=True AND FLAG=True AND LUZ_INT[i]=False AND ((PARADO=True AND PLANTA<>i) OR PARADO=False) THEN
      LUZ_INT[i]:=True;
      IF (i<PLANTA AND SUBIR=False) OR (i<=PLANTA AND SUBIR=True) THEN
        TURNO_BAJ[i]:=1;
      ELSIF (i>PLANTA AND BAJAR=False) OR (i>=PLANTA AND BAJAR=True) THEN
        TURNO_SUB[i]:=1;
      END_IF;
      IF OCUPADO=False AND AT8=False THEN
        PISO_PULSADO:=i;
        AUXILIAR:=i;
      END_IF;
      OCUPADO:=True;
    END_IF;
  END_FOR;
END_IF;

```

```

IF PISO_PULSADO<>0 THEN
  IF TURNO_SUB[PISO_PULSADO]=0 AND ACABA_CERRAR=TRUE AND Vengo_de_abajo=TRUE AND RECOGIDA_BAJAR=FALSE THEN
    IF LUZ_BAJ[PISO_PULSADO]=FALSE AND LUZ_SUB[PISO_PULSADO]=FALSE AND TURNO_BAJ[PISO_PULSADO]=FALSE THEN
      FOR i:=PISO_PULSADO TO 4 BY 1 DO
        IF TURNO_SUB[i]=1 AND (LUZ_INT[i]=TRUE OR PISO_PULSADO2<>i OR STOP2=TRUE) THEN
          PISO_PULSADO:=i;
          i:=5;
        END_IF;
      END_FOR;
      IF AUXILIAR=PISO_PULSADO THEN
        FOR i:=4 TO 1 BY -1 DO
          IF TURNO_BAJ[i]=1 AND (LUZ_INT[i]=TRUE OR PISO_PULSADO2<>i OR STOP2=TRUE) THEN
            IF i>PISO_PULSADO THEN
              RECOGIDA_BAJAR:=TRUE;
            END_IF;
            PISO_PULSADO:=i;
            i:=0;
          END_IF;
        END_FOR;
        IF AUXILIAR=PISO_PULSADO THEN
          FOR i:=1 TO PISO_PULSADO BY 1 DO
            IF TURNO_SUB[i]=1 AND (LUZ_INT[i]=TRUE OR PISO_PULSADO2<>i OR STOP2=TRUE) THEN
              PISO_PULSADO:=i;
              RECOGIDA_SUBIR:=TRUE;
              i:=5;
            END_IF;
          END_FOR;
          IF AUXILIAR=PISO_PULSADO THEN
            PISO_PULSADO:=0;
          END_IF;
        END_IF;
      END_IF;
      AUXILIAR:=PISO_PULSADO;
    END_IF;
  END_IF;
END_IF;

IF PISO_PULSADO<>0 THEN
  IF TURNO_BAJ[PISO_PULSADO]=0 AND ACABA_CERRAR=TRUE AND Vengo_de_arriba=TRUE AND RECOGIDA_SUBIR=FALSE THEN
    IF LUZ_BAJ[PISO_PULSADO]=FALSE AND LUZ_SUB[PISO_PULSADO]=FALSE AND TURNO_SUB[PISO_PULSADO]=FALSE THEN
      FOR i:=PISO_PULSADO TO 1 BY -1 DO
        IF TURNO_BAJ[i]=1 AND (LUZ_INT[i]=TRUE OR PISO_PULSADO2<>i OR STOP2=TRUE) THEN
          PISO_PULSADO:=i;
          i:=0;
        END_IF;
      END_FOR;
      IF AUXILIAR=PISO_PULSADO THEN
        FOR i:=1 TO 4 BY 1 DO
          IF TURNO_SUB[i]=1 AND (LUZ_INT[i]=TRUE OR PISO_PULSADO2<>i OR STOP2=TRUE) THEN
            IF i<PISO_PULSADO THEN
              RECOGIDA_SUBIR:=TRUE;
            END_IF;
            PISO_PULSADO:=i;
            i:=5;
          END_IF;
        END_FOR;
        IF AUXILIAR=PISO_PULSADO THEN
          FOR i:=4 TO PISO_PULSADO BY -1 DO
            IF TURNO_BAJ[i]=1 AND (LUZ_INT[i]=TRUE OR PISO_PULSADO2<>i OR STOP2=TRUE) THEN
              PISO_PULSADO:=i;
              RECOGIDA_BAJAR:=TRUE;
              i:=0;
            END_IF;
          END_FOR;
          IF AUXILIAR=PISO_PULSADO THEN
            PISO_PULSADO:=0;
          END_IF;
        END_IF;
      END_IF;
      AUXILIAR:=PISO_PULSADO;
    END_IF;
  END_IF;
END_IF;

IF PISO_PULSADO<>0 THEN
  IF TURNO_SUB[PISO_PULSADO]=0 AND ACABA_CERRAR=TRUE AND Vengo_de_arriba=TRUE AND RECOGIDA_SUBIR=TRUE THEN
    IF LUZ_BAJ[PISO_PULSADO]=FALSE AND LUZ_SUB[PISO_PULSADO]=FALSE AND TURNO_BAJ[PISO_PULSADO]=FALSE THEN
      RECOGIDA_BAJAR:=FALSE;
      FOR i:=4 TO PISO_PULSADO BY -1 DO
        IF TURNO_SUB[i]=1 AND (LUZ_INT[i]=TRUE OR PISO_PULSADO2<>i OR STOP2=TRUE) THEN
          PISO_PULSADO:=i;
          RECOGIDA_SUBIR:=FALSE;
          i:=0;
        END_IF;
      END_FOR;
      IF AUXILIAR=PISO_PULSADO THEN
        FOR i:=PISO_PULSADO TO 1 BY -1 DO
          IF TURNO_BAJ[i]=1 AND (LUZ_INT[i]=TRUE OR PISO_PULSADO2<>i OR STOP2=TRUE) THEN
            PISO_PULSADO:=i;
            RECOGIDA_SUBIR:=FALSE;
            i:=0;
          END_IF;
        END_FOR;
        IF AUXILIAR=PISO_PULSADO THEN
          FOR i:=1 TO PISO_PULSADO BY 1 DO
            IF TURNO_SUB[i]=1 AND (LUZ_INT[i]=TRUE OR PISO_PULSADO2<>i OR STOP2=TRUE) THEN
              PISO_PULSADO:=i;
              i:=5;
            END_IF;
          END_FOR;
          IF AUXILIAR=PISO_PULSADO THEN
            FOR i:=4 TO PISO_PULSADO BY -1 DO
              IF TURNO_BAJ[i]=1 AND (LUZ_INT[i]=TRUE OR PISO_PULSADO2<>i OR STOP2=TRUE) THEN
                PISO_PULSADO:=i;
                RECOGIDA_BAJAR:=TRUE;
                RECOGIDA_SUBIR:=FALSE;
                i:=0;
              END_IF;
            END_FOR;
            IF AUXILIAR=PISO_PULSADO THEN
              PISO_PULSADO:=0;
              RECOGIDA_SUBIR:=FALSE;
            END_IF;
          END_IF;
        END_IF;
      END_IF;
      AUXILIAR:=PISO_PULSADO;
    END_IF;
  END_IF;
END_IF;

```



```

IF PISO_PULSADO<>0 THEN
  IF TURNO_BAJ[PISO_PULSADO]=0 AND ACABA_CERRAR=TRUE AND Vengo_de_abajo=TRUE AND RECOGIDA_BAJAR=TRUE THEN
    IF LUZ_BAJ[PISO_PULSADO]=FALSE AND LUZ_SUB[PISO_PULSADO]=FALSE AND TURNO_SUB[PISO_PULSADO]=FALSE THEN
      RECOGIDA_SUBIR:=FALSE;
      FOR i:=1 TO PISO_PULSADO BY 1 DO
        IF TURNO_BAJ[i]=1 AND (LUZ_INT[i]=TRUE OR PISO_PULSADO<>i OR STOP2=TRUE) THEN
          PISO_PULSADO:=i;
          RECOGIDA_BAJAR:=FALSE;
          i:=5;
        END_IF;
      END_FOR;
      IF AUXILIAR=PISO_PULSADO THEN
        FOR i:=PISO_PULSADO TO 4 BY 1 DO
          IF TURNO_SUB[i]=1 AND (LUZ_INT[i]=TRUE OR PISO_PULSADO2<>i OR STOP2=TRUE) THEN
            PISO_PULSADO:=i;
            RECOGIDA_BAJAR:=FALSE;
            i:=5;
          END_IF;
        END_FOR;
        IF AUXILIAR=PISO_PULSADO THEN
          FOR i:=4 TO PISO_PULSADO BY -1 DO
            IF TURNO_BAJ[i]=1 AND (LUZ_INT[i]=TRUE OR PISO_PULSADO2<>i OR STOP2=TRUE) THEN
              PISO_PULSADO:=i;
              i:=0;
            END_IF;
          END_FOR;
          IF AUXILIAR=PISO_PULSADO THEN
            FOR i:=1 TO PISO_PULSADO BY 1 DO
              IF TURNO_SUB[i]=1 AND (LUZ_INT[i]=TRUE OR PISO_PULSADO2<>i OR STOP2=TRUE) THEN
                PISO_PULSADO:=i;
                RECOGIDA_SUBIR:=TRUE;
                RECOGIDA_BAJAR:=FALSE;
                i:=5;
              END_IF;
            END_FOR;
            IF AUXILIAR=PISO_PULSADO THEN
              PISO_PULSADO:=0;
              RECOGIDA_BAJAR:=FALSE;
            END_IF;
          END_IF;
        END_IF;
      END_IF;
      AUXILIAR:=PISO_PULSADO;
    END_IF;
  END_IF;
END_IF;

FOR i:=1 TO 4 BY 1 DO
  IF ACTUALIZA=TRUE AND SENSOR[i]=TRUE THEN
    LUZ_INT[i]:=FALSE;
    TURNO_SUB[i]:=0;
    LUZ_SUB[i]:=0;
    TURNO_BAJ[i]:=0;
    LUZ_BAJ[i]:=0;
    IF PISO_PULSADO2=i AND LUZ_INT2[i]=FALSE THEN
      IF Subir2=TRUE AND RECOGIDA_BAJAR2=FALSE THEN
        FOR j:=4 TO PLANTA2 BY -1 DO
          IF TURNO2_SUB[j]=1 AND j<>i THEN
            TURNO2_SUB[PISO_PULSADO2]:=0;
            TURNO2_BAJ[PISO_PULSADO2]:=0;
            PISO_PULSADO2:=j;
            j:=0;
          END_IF;
        END_FOR;
      ELSEIF Bajar2=TRUE AND RECOGIDA_SUBIR2=FALSE THEN
        FOR j:=PLANTA2 TO 1 BY -1 DO
          IF TURNO2_BAJ[j]=1 AND j<>i THEN
            TURNO2_BAJ[PISO_PULSADO2]:=0;
            TURNO2_SUB[PISO_PULSADO2]:=0;
            PISO_PULSADO2:=j;
            j:=0;
          END_IF;
        END_FOR;
      END_IF;
      IF (PISO_PULSADO2<>i OR (SUBIR2=FALSE AND BAJAR2=FALSE)) AND LUZ_INT2[i]=FALSE THEN
        IF TURNO_SUB[i]=0 THEN
          TURNO2_SUB[i]:=0;
        END_IF;
        IF TURNO_BAJ[i]=0 THEN
          TURNO2_BAJ[i]:=0;
        END_IF;
      END_IF;
    END_IF;
  END_FOR;

FOR i:=1 TO 4 BY 1 DO
  IF SENSOR[i]=TRUE AND RECOGIDA_SUBIR=FALSE AND RECOGIDA_BAJAR=FALSE AND
  ((SUBIR=TRUE AND TURNO_SUB[i]=1) OR (BAJAR=TRUE AND TURNO_BAJ[i]=1)) THEN
    ABRE:=TRUE;
  END_IF;
END_FOR;

IF SIEMPRE=TRUE THEN
  CONT:=0;
  FOR i:=1 TO 4 BY 1 DO
    IF TURNO_SUB[i]<>0 OR TURNO_BAJ[i]<>0 THEN
      CONT:=CONT+1;
    END_IF;
  END_FOR;
  IF CONT=0 THEN
    OCUPADO:=FALSE;
  ELSE
    OCUPADO:=TRUE;
  END_IF;
END_IF;

```

```

(**ASCENSOR 2**)

IF STOP2=FALSE THEN
  FOR i:=1 TO 4 BY 1 DO
    IF PULSADOR_INT2[i]=TRUE AND FLAG2=TRUE AND LUZ_INT2[i]=FALSE AND ((PARADO2=TRUE AND PLANTA2<>i) OR PARADO2=FALSE) THEN
      LUZ_INT2[i]:=TRUE;
      IF OCUPADO2=FALSE AND AT82=FALSE THEN
        PISO_PULSADO2:=i;
        AUXILIAR2:=i;
      END_IF;
      IF (i<PLANTA2 AND SUBIR2=FALSE) OR (i<=PLANTA2 AND SUBIR2=TRUE) THEN
        TURNO2_BAJ[i]:=1;
      ELSEIF (i>PLANTA2 AND BAJAR2=FALSE) OR (i>=PLANTA2 AND BAJAR2=TRUE) THEN
        TURNO2_SUB[i]:=1;
      END_IF;
      OCUPADO2:=TRUE;
    END_IF;
  END_FOR;
END_IF;

IF PISO_PULSADO2<>0 THEN
  IF TURNO2_SUB[PISO_PULSADO2]=0 AND ACABA_CERRAR2=TRUE AND Vengo_de_abaajo2=TRUE AND RECOGIDA_BAJAR2=FALSE THEN
    IF LUZ_BAJ[PISO_PULSADO2]=FALSE AND LUZ_SUB[PISO_PULSADO2]=FALSE AND TURNO2_BAJ[PISO_PULSADO2]=FALSE THEN
      FOR i:=PISO_PULSADO2 TO 4 BY 1 DO
        IF TURNO2_SUB[i]=1 AND (LUZ_INT2[i]=TRUE OR PISO_PULSADO2<>i OR STOP1=TRUE) THEN
          PISO_PULSADO2:=i;
          i:=5;
        END_IF;
      END_FOR;
      IF AUXILIAR2=PISO_PULSADO2 THEN
        FOR i:=4 TO 1 BY -1 DO
          IF TURNO2_BAJ[i]=1 AND (LUZ_INT2[i]=TRUE OR PISO_PULSADO2<>i OR STOP1=TRUE) THEN
            IF i>PISO_PULSADO2 THEN
              RECOGIDA_BAJAR2:=TRUE;
            END_IF;
            PISO_PULSADO2:=i;
            i:=0;
          END_IF;
        END_FOR;
        IF AUXILIAR2=PISO_PULSADO2 THEN
          FOR i:=1 TO PISO_PULSADO2 BY 1 DO
            IF TURNO2_SUB[i]=1 AND (LUZ_INT2[i]=TRUE OR PISO_PULSADO2<>i OR STOP1=TRUE) THEN
              PISO_PULSADO2:=i;
              RECOGIDA_SUBIR2:=TRUE;
              i:=5;
            END_IF;
          END_FOR;
          IF AUXILIAR2=PISO_PULSADO2 THEN
            PISO_PULSADO2:=0;
          END_IF;
        END_IF;
      END_IF;
      AUXILIAR2:=PISO_PULSADO2;
    END_IF;
  END_IF;

  IF PISO_PULSADO2<>0 THEN
    IF TURNO2_BAJ[PISO_PULSADO2]=0 AND ACABA_CERRAR2=TRUE AND Vengo_de_arriba2=TRUE AND RECOGIDA_SUBIR2=FALSE THEN
      IF LUZ_BAJ[PISO_PULSADO2]=FALSE AND LUZ_SUB[PISO_PULSADO2]=FALSE AND TURNO2_SUB[PISO_PULSADO2]=FALSE THEN
        FOR i:=PISO_PULSADO2 TO 1 BY -1 DO
          IF TURNO2_BAJ[i]=1 AND (LUZ_INT2[i]=TRUE OR PISO_PULSADO2<>i OR STOP1=TRUE) THEN
            PISO_PULSADO2:=i;
            i:=0;
          END_IF;
        END_FOR;
        IF AUXILIAR2=PISO_PULSADO2 THEN
          FOR i:=1 TO 4 BY 1 DO
            IF TURNO2_SUB[i]=1 AND (LUZ_INT2[i]=TRUE OR PISO_PULSADO2<>i OR STOP1=TRUE) THEN
              IF i<PISO_PULSADO2 THEN
                RECOGIDA_SUBIR2:=TRUE;
              END_IF;
              PISO_PULSADO2:=i;
              i:=5;
            END_IF;
          END_FOR;
          IF AUXILIAR2=PISO_PULSADO2 THEN
            FOR i:=4 TO PISO_PULSADO2 BY -1 DO
              IF TURNO2_BAJ[i]=1 AND (LUZ_INT2[i]=TRUE OR PISO_PULSADO2<>i OR STOP1=TRUE) THEN
                PISO_PULSADO2:=i;
                RECOGIDA_BAJAR2:=TRUE;
                i:=0;
              END_IF;
            END_FOR;
            IF AUXILIAR2=PISO_PULSADO2 THEN
              PISO_PULSADO2:=0;
            END_IF;
          END_IF;
        END_IF;
      END_IF;
      AUXILIAR2:=PISO_PULSADO2;
    END_IF;
  END_IF;
END_IF;

```

```

IF PISO_PULSADO2<>0 THEN
  IF TURNO2_SUB[PISO_PULSADO2]=0 AND ACABA_CERRAR2=TRUE AND Vengo_de_arriba2=TRUE AND RECOGIDA_SUBIR2=TRUE THEN
    IF LUZ_BAJ[PISO_PULSADO2]=FALSE AND LUZ_SUB[PISO_PULSADO2]=FALSE AND TURNO2_BAJ[PISO_PULSADO2]=FALSE THEN
      RECOGIDA_BAJAR2:=FALSE;
      FOR i:=4 TC PISO_PULSADO2 BY -1 DC
        IF TURNO2_SUB[i]=1 AND (LUZ_INT2[i]=TRUE OR PISO_PULSADO2<>i OR STOP1=TRUE) THEN
          PISO_PULSADO2:=i;
          RECOGIDA_SUBIR2:=FALSE;
          i:=0;
        END_IF;
      END_FOR;
      IF AUXILIAR2=PISO_PULSADO2 THEN
        FOR i:=PISO_PULSADO2 TC 1 BY -1 DC
          IF TURNO2_BAJ[i]=1 AND (LUZ_INT2[i]=TRUE OR PISO_PULSADO2<>i OR STOP1=TRUE) THEN
            PISO_PULSADO2:=i;
            RECOGIDA_SUBIR2:=FALSE;
            i:=0;
          END_IF;
        END_FOR;
        IF AUXILIAR2=PISO_PULSADO2 THEN
          FOR i:=1 TC PISO_PULSADO2 BY 1 DC
            IF TURNO2_SUB[i]=1 AND (LUZ_INT2[i]=TRUE OR PISO_PULSADO2<>i OR STOP1=TRUE) THEN
              PISO_PULSADO2:=i;
              i:=5;
            END_IF;
          END_FOR;
          IF AUXILIAR2=PISO_PULSADO2 THEN
            FOR i:=4 TC PISO_PULSADO2 BY -1 DC
              IF TURNO2_BAJ[i]=1 AND (LUZ_INT2[i]=TRUE OR PISO_PULSADO2<>i OR STOP1=TRUE) THEN
                PISO_PULSADO2:=i;
                RECOGIDA_BAJAR2:=TRUE;
                RECOGIDA_SUBIR2:=FALSE;
                i:=0;
              END_IF;
            END_FOR;
            IF AUXILIAR2=PISO_PULSADO2 THEN
              PISO_PULSADO2:=0;
              RECOGIDA_SUBIR2:=FALSE;
            END_IF;
          END_IF;
        END_IF;
      END_IF;
      AUXILIAR2:=PISO_PULSADO2;
    END_IF;
  END_IF;
END_IF;

IF PISO_PULSADO2<>0 THEN
  IF TURNO2_BAJ[PISO_PULSADO2]=0 AND ACABA_CERRAR2=TRUE AND Vengo_de_abajo2=TRUE AND RECOGIDA_BAJAR2=TRUE THEN
    IF LUZ_BAJ[PISO_PULSADO2]=FALSE AND LUZ_SUB[PISO_PULSADO2]=FALSE AND TURNO2_SUB[PISO_PULSADO2]=FALSE THEN
      RECOGIDA_SUBIR2:=FALSE;
      FOR i:=1 TC PISO_PULSADO2 BY 1 DC
        IF TURNO2_BAJ[i]=1 AND (LUZ_INT2[i]=TRUE OR PISO_PULSADO2<>i OR STOP1=TRUE) THEN
          PISO_PULSADO2:=i;
          RECOGIDA_BAJAR2:=FALSE;
          i:=5;
        END_IF;
      END_FOR;
      IF AUXILIAR2=PISO_PULSADO2 THEN
        FOR i:=PISO_PULSADO2 TC 4 BY 1 DC
          IF TURNO2_SUB[i]=1 AND (LUZ_INT2[i]=TRUE OR PISO_PULSADO2<>i OR STOP1=TRUE) THEN
            PISO_PULSADO2:=i;
            RECOGIDA_BAJAR2:=FALSE;
            i:=5;
          END_IF;
        END_FOR;
        IF AUXILIAR2=PISO_PULSADO2 THEN
          FOR i:=4 TC PISO_PULSADO2 BY -1 DC
            IF TURNO2_BAJ[i]=1 AND (LUZ_INT2[i]=TRUE OR PISO_PULSADO2<>i OR STOP1=TRUE) THEN
              PISO_PULSADO2:=i;
              i:=0;
            END_IF;
          END_FOR;
          IF AUXILIAR2=PISO_PULSADO2 THEN
            FOR i:=1 TC PISO_PULSADO2 BY 1 DC
              IF TURNO2_SUB[i]=1 AND (LUZ_INT2[i]=TRUE OR PISO_PULSADO2<>i OR STOP1=TRUE) THEN
                PISO_PULSADO2:=i;
                RECOGIDA_SUBIR2:=TRUE;
                RECOGIDA_BAJAR2:=FALSE;
                i:=5;
              END_IF;
            END_FOR;
            IF AUXILIAR2=PISO_PULSADO2 THEN
              PISO_PULSADO2:=0;
              RECOGIDA_BAJAR2:=FALSE;
            END_IF;
          END_IF;
        END_IF;
      END_IF;
      AUXILIAR2:=PISO_PULSADO2;
    END_IF;
  END_IF;
END_IF;

```



```

FOR i:=1 TO 4 BY 1 DO
  IF ACTUALIZA2=TRUE AND SENSOR2[i]=TRUE THEN
    LUZ_INT2[i]:=FALSE;
    TURNO2_BAJ[i]:=0;
    LUZ_BAJ[i]:=0;
    TURNO2_SUB[i]:=0;
    LUZ_SUB[i]:=0;
    IF PISO_PULSADO=i AND LUZ_INT[i]=FALSE THEN
      IF Subir=TRUE AND RECOGIDA_BAJAR=FALSE THEN
        FOR j:=4 TO PLANTA BY -1 DO
          IF TURNO_SUB[j]=1 AND j<>i THEN
            TURNO_SUB[PISO_PULSADO]:=0;
            PISO_PULSADO:=j;
            j:=0;
          END IF;
        END FOR;
      ELSEIF Bajar=TRUE AND RECOGIDA_SUBIR=FALSE THEN
        FOR j:=PLANTA TO 1 BY -1 DO
          IF TURNO_BAJ[j]=1 AND j<>i THEN
            TURNO_BAJ[PISO_PULSADO]:=0;
            PISO_PULSADO:=j;
            j:=0;
          END IF;
        END FOR;
      END IF;
    ELSEIF (PISO_PULSADO<>i OR (SUBIR=FALSE AND BAJAR=FALSE)) AND LUZ_INT[i]=FALSE THEN
      IF TURNO2_SUB[i]=0 THEN
        TURNO_SUB[i]:=0;
      END IF;
      IF TURNO2_BAJ[i]=0 THEN
        TURNO_BAJ[i]:=0;
      END IF;
    END IF;
  END IF;
END FOR;

FOR i:=1 TO 4 BY 1 DO
  IF SENSOR2[i]=TRUE AND RECOGIDA_SUBIR2=FALSE AND RECOGIDA_BAJAR2=FALSE AND
    ((SUBIR2=TRUE AND TURNO2_SUB[i]=1) OR (BAJAR2=TRUE AND TURNO2_BAJ[i]=1)) THEN
    ABRE2:=TRUE;
  END IF;
END FOR;

IF SIEMPRE=TRUE THEN
  CONT2:=0;
  FOR i:=1 TO 4 BY 1 DO
    IF TURNO2_SUB[i]<>0 OR TURNO2_BAJ[i]<>0 THEN
      CONT2:=CONT2+1;
    END IF;
  END FOR;
  IF CONT2=0 THEN
    OCUPADO2:=FALSE;
  ELSE
    OCUPADO2:=TRUE;
  END IF;
END IF;

FOR i:=1 TO 4 BY 1 DO
  IF PISO_PULSADO=i AND LUZ_INT2[i]=FALSE AND PISO_PULSADO2<>i AND ((RECOGIDA_SUBIR=FALSE AND
    RECOGIDA_BAJAR=FALSE) OR ((RECOGIDA_SUBIR=TRUE OR RECOGIDA_BAJAR=TRUE) AND CONT2=1)) THEN
    TURNO2_SUB[i]:=FALSE;
    TURNO2_BAJ[i]:=FALSE;
  ELSEIF PISO_PULSADO2=i AND LUZ_INT[i]=FALSE AND PISO_PULSADO<>i AND ((RECOGIDA_SUBIR2=FALSE AND
    RECOGIDA_BAJAR2=FALSE) OR ((RECOGIDA_SUBIR2=TRUE OR RECOGIDA_BAJAR2=TRUE) AND CONT2=1)) THEN
    TURNO_SUB[i]:=FALSE;
    TURNO_BAJ[i]:=FALSE;
  END IF;
END FOR;

IF STOP1=TRUE AND STOP2=TRUE THEN
  FOR i:=4 TO 1 BY -1 DO
    TURNO_SUB[i]:=0;
    TURNO_BAJ[i]:=0;
    TURNO2_SUB[i]:=0;
    TURNO2_BAJ[i]:=0;
    LUZ_INT[i]:=0;
    LUZ_INT2[i]:=0;
    LUZ_SUB[i]:=0;
    LUZ_BAJ[i]:=0;
    OCUPADO:=FALSE;
    OCUPADO2:=FALSE;
  END FOR;
END IF;

```

PARADAS (ST)

```

IF STOP1=TRUE AND STOP2=FALSE THEN
  FOR i:=1 TO 4 BY 1 DO
    IF TURNO2_SUB[i]=0 AND ((PULSADOR_SUB[i]=1 AND LUZ_SUB[i]=FALSE) OR (LUZ_SUB[i]=TRUE AND FT10=TRUE)) THEN
      IF OCUPADO2=FALSE THEN
        PISO_PULSADO2:=i;
        AUXILIAR2:=i;
        OCUPADO2:=TRUE;
        IF i<PLANTA2 THEN
          RECOGIDA_SUBIR2:=TRUE;
        END_IF;
      END_IF;
      LUZ_SUB[i]:=TRUE;
      TURNO2_SUB[i]:=1;
      TURNO2_SUB[i]:=0;
    END_IF;
  END_FOR;
  IF TURNO2_BAJ[i]=0 AND ((PULSADOR_BAJ[i]=1 AND LUZ_BAJ[i]=FALSE) OR (LUZ_BAJ[i]=TRUE AND FT10=TRUE)) THEN
    IF OCUPADO2=FALSE THEN
      PISO_PULSADO2:=i;
      AUXILIAR2:=i;
      OCUPADO2:=TRUE;
      IF i>PLANTA2 THEN
        RECOGIDA_BAJAR2:=TRUE;
      END_IF;
    END_IF;
    LUZ_BAJ[i]:=TRUE;
    TURNO2_BAJ[i]:=1;
    TURNO2_BAJ[i]:=0;
  END_IF;
END_IF;

IF STOP2=TRUE AND STOP1=FALSE THEN
  FOR i:=1 TO 4 BY 1 DO
    IF TURNO_SUB[i]=0 AND ((PULSADOR_SUB[i]=1 AND LUZ_SUB[i]=FALSE) OR (LUZ_SUB[i]=TRUE AND FT10=TRUE)) THEN
      IF OCUPADO=FALSE THEN
        PISO_PULSADC:=i;
        AUXILIAR:=i;
        OCUPADO:=TRUE;
        IF i<PLANTA THEN
          RECOGIDA_SUBIR:=TRUE;
        END_IF;
      END_IF;
      LUZ_SUB[i]:=TRUE;
      TURNO_SUB[i]:=1;
      TURNO2_SUB[i]:=0;
    END_IF;
  END_FOR;
  IF TURNO_BAJ[i]=0 AND ((PULSADOR_BAJ[i]=1 AND LUZ_BAJ[i]=FALSE) OR (LUZ_BAJ[i]=TRUE AND FT10=TRUE)) THEN
    IF OCUPADO=FALSE THEN
      PISO_PULSADC:=i;
      AUXILIAR:=i;
      OCUPADO:=TRUE;
      IF i>PLANTA THEN
        RECOGIDA_BAJAR:=TRUE;
      END_IF;
    END_IF;
    LUZ_BAJ[i]:=TRUE;
    TURNO_BAJ[i]:=1;
    TURNO2_BAJ[i]:=0;
  END_IF;
END_FOR;
END_IF;

IF LIBRE=TRUE THEN
  RECOLOCA:=TRUE;
  TURNO_BAJ[1]:=1;
  PISO_PULSADC:=1;
  OCUPADO:=TRUE;
  AUXILIAR:=1;
END_IF;

```



```

IF LIBRE2=TRUE THEN
    RECOLOCA2:=TRUE;
    TURNO2_SUB[4]:=1;
    PISO_PULSADO2:=4;
    OCUPADO2:=TRUE;
    AUXILIAR2:=4;
END_IF;

IF ACABA_CERRAR=TRUE AND RECOLOCA=TRUE THEN
    RECOLOCA:=FALSE;
    IF LUZ_INT[1]=0 THEN
        TURNO_BAJ[1]:=0;
    END_IF;
END_IF;

IF ACABA_CERRAR2=TRUE AND RECOLOCA2=TRUE THEN
    RECOLOCA2:=FALSE;
    IF LUZ_INT2[4]=0 THEN
        TURNO2_SUB[4]:=0;
    END_IF;
END_IF;

(*ASCENSOR 1*)

FOR i:=1 TO 4 BY 1 DO
    IF SENSOR[i]=1 THEN
        PLANTA:=i;
    END_IF;
END_FOR;

IF PISO_PULSADO>PLANTA THEN
    MAYOR:=1;
    MENOR:=0;
    IGUAL:=0;
ELSIF PLANTA>PISO_PULSADO AND PISO_PULSADO<>0 THEN
    MENOR:=1;
    MAYOR:=0;
    IGUAL:=0;
ELSE
    IGUAL:=1;
    MAYOR:=0;
    MENOR:=0;
END_IF;

(*ASCENSOR 2*)

FOR i:=1 TO 4 BY 1 DO
    IF SENSOR2[i]=1 THEN
        PLANTA2:=i;
    END_IF;
END_FOR;

IF PISO_PULSADO2>PLANTA2 THEN
    MAYOR2:=1;
    MENOR2:=0;
    IGUAL2:=0;
ELSIF PLANTA2>PISO_PULSADO2 AND PISO_PULSADO2<>0 THEN
    MENOR2:=1;
    MAYOR2:=0;
    IGUAL2:=0;
ELSE
    IGUAL2:=1;
    MAYOR2:=0;
    MENOR2:=0;
END_IF;

```


REFERENCIAS

- [1] RAE: *Definición ascensor*, <http://dle.rae.es/srv/fetch?id=3vosANs>
- [2] Prieto, M. J.: *El ascensor de Nerón*, <http://www.curistoria.com/2009/01/el-ascensor-de-nern.html>
- [3] *Elisha Graves Otis*, Biografías y Vidas, https://www.biografiasyvidas.com/biografia/o/otis_elisha.htm
- [4] *Antigüedad eléctrica, Historia del ascensor*, <http://www.afinidadelectrica.com/articulo.php?IdArticulo=125>
- [5] *Un poco de historia del ascensor*, Silcon Electrónica, http://www.silcon.com.ar/un_poco_de_historia.htm
- [6] *Datos históricos del ascensor*, Mitsubishi Electric Corporation, <http://www.mitsubishielectric.com/elevator/es/overview/elevators/history.html>
- [7] Miravete, A., Larrodé E.: *Elevadores: Principios e innovaciones*, Ed. Reverté, S.A., Cap. 2.1, 2007
- [8] *Ascensor eléctrico*, <http://www.ascensoresrekalde.com/pag.php?id=12&pag=ascensor-electrico>
- [9] *Ascensor sin cuarto de máquinas*, <https://arquigrafico.com/ascensores-sin-cuarto-de-maquinas/>
- [10] *Diferencias entre un ascensor eléctrico y uno hidráulico*, Citylift Ascensores, <https://cityliftascensores.com/diferencias-entre-un-ascensor-electrico-y-uno-hidraulico/>
- [11] *Tipos de Ascensores para las edificaciones*, <https://arquigrafico.com/tipos-de-ascensores-para-las-edificaciones/>
- [12] *Maniobras de ascensores*, Colegio Oficial de Arquitectos de Galicia, Universidade da Coruña
- [13] *Maniobra colectiva en ascensores: Cómo ahorrar*, <http://www.cume24.es/maniobra-colectiva-en-ascensores-como-ahorrar/>
- [14] Miravete, A., Larrodé E.: *Elevadores: Principios e innovaciones*, Ed. Reverté, S.A., Cap. 2.2, 2007
- [15] *Ascensores*, Universidad de Salamanca, <http://dim.usal.es/areaim/guia%20P.%20I/ascensores.htm>
- [16] *Limitador de velocidad*, Dinattech, http://www.dynattech-elevation.com/documentos/articulos/documentos_REDUCIDO_VEGA-ESP_820d9d46.pdf
- [17] *Cuadro de maniobra de los ascensores*, <http://a3ascensores.es/cuadro-maniobras-los-ascensores/>
- [18] *Partes del ascensor*, Eninter Ascensores, <http://www.eninter.com/blog/partes-del-ascensor/>
- [19] *El contrapeso de un ascensor*, Obraxa. Instalación de ascensores, <http://www.instalacionascensoresvalencia.es/el-contrapeso-de-un-ascensor/>
- [20] *Real Decreto 203/216, Requisitos esenciales de seguridad para la comercialización de ascensores y componentes de seguridad para ascensores*, BOE, <https://www.boe.es/boe/dias/2016/05/25/pdfs/BOE-A-2016-4953.pdf>
- [21] Pascual, J., *Componentes de seguridad en un ascensor eléctrico*, 2014, <http://ascensoresjpascual.blogspot.com.es/2014/03/componentes-de-seguridad-en-un-ascensor.html>

- [22] *Modicon M340 con Unity Pro. Procesadores, bastidores y módulos de fuente de alimentación. Manual de configuración*, Schneider Electric, págs. 75, 79-92, 124-133 y 170-173, 2009.
- [23] *Modicon M340 con Unity Pro. Módulos de entradas/salidas binarias. Manual de usuario*, Schneider Electric, págs. 119-124 y 183-192, 2012
- [24] *Codesys*, <https://www.codesys.com/>
- [25] *Controladores Lógicos Programables (PLCs)*, Departamento de Ingeniería Eléctrica, Electrónica y Control (IEEC), Universidad Nacional de Estudios a Distancia (INED), http://www.ieec.uned.es/investigacion/Dipseil/PAC/archivos/Informacion_de_referencia_ISE6_1_2.pdf
- [26] Cambroner, A., *Programación en ST*, OMRON, 2009, http://instrumentacionycontrol.net/Descargas/Descargas/OMRON/CX-One_Cx-Programmer_Syswin/IyCnet_Omron_Programacion_ST.pdf
- [27] *Lenguaje de programación S7, CFC*, <http://www.sc.ehu.es/sbweb/webcentro/automatica/Step7/paginas/contenido/step7/7/2.9.1.7.htm>
- [28] *Introducción al software de programación Codesys*, Laboratorio Remoto de Automática (LRA-ULE), Universidad de León, http://lra.unileon.es/sites/lra.unileon.es/files/Documents/plc/Moeller/Manual%20Software_Codesys.pdf
- [29] *Unity Pro, Software de configuración para Modicon M340, Premium y Quantum*, Schneider Electric, <https://www.schneider-electric.es/es/product-range-presentation/548-unity-pro#tabs-top>
- [30] Instituto Schneider Electric de Formación, *Manual Unity Pro*, Schneider Electric España, 2008, http://www.equiposdidacticos.com/pdf/catalogos/manual_unity.pdf

GLOSARIO

RAE: Real Academia Española	1
a.C.: antes de Cristo	3
UNE: Una Norma Española	10
PLC: Programmable Logic Controller (Controlador Lógico Programable)	12
VCC: Voltaje de Corriente Continua	17
CPU: Central Processing Unit (Unidad Central de Proceso)	20
DSP: Digital Signal Processor (Procesador Digital de Señales)	33
IEC: International Electrotechnical Commission (Comisión Electrotécnica Internacional)	33
POU: Program Organization Unit (Unidad Organizativa de Programa)	33
SFC: Sequential Function Chart (Diagrama de Funciones Secuenciales)	33
FBD: Function Block Diagram (Diagrama de Bloques de Funciones)	33
LD: Ladder (Diagrama de contactos)	33
ST: Structured Text (Texto Estructurado)	33
IL: Instructions List (Lista de Instrucciones)	33
CFC: Continuous Function Chart (Diagrama de Funciones Continuas)	33